



**Contributions dans le domaine de l'analyse
multirésolution de maillages surfaciques semi-réguliers.
Application à la compression géométrique**

Aymen Kammoun

► **To cite this version:**

Aymen Kammoun. Contributions dans le domaine de l'analyse multirésolution de maillages surfaciques semi-réguliers. Application à la compression géométrique. Traitement du signal et de l'image [eess.SP]. Université de Nice-Sophia Antipolis, 2011. Français. NNT: . tel-01355283

HAL Id: tel-01355283

<https://hal.science/tel-01355283>

Submitted on 23 Aug 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE DE NICE-SOPHIA ANTIPOLIS

ECOLE DOCTORALE STIC
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

T H E S E

pour obtenir le titre de

Docteur en Sciences

de l'Université de Nice-Sophia Antipolis

Mention : Automatique, Traitement du Signal et des Images

présentée par
Aymen KAMMOUN

Le 12 décembre 2011

**Contributions dans le domaine de l'analyse multirésolution de
maillages surfaciques semi-réguliers. Application à la compression
géométrique**

Thèse dirigée par Marc ANTONINI

Jury :

M. Mohamed DAOUDI	Professeur	Président du jury
Mme. Amel BENAZZA	Professeur	Rapporteur
M. Titus ZAHARIA	Professeur	Rapporteur
M. Marc ANTONINI	Directeur de recherche	Directeur de thèse
M. Frédéric PAYAN	Maître de conférence	Co-encadrant
M. Basile SAUVAGE	Maître de conférence	Examineur

*A mon père
Mohamed,*

*à ma mère
Samira,*

*à mon frère
Akram,*

*à ma sœur
Amal,*

*à mes parents
et
amis.*

Table des matières

Table des matières	i
Introduction générale	3
I Maillages statiques	9
Introduction	11
1 Modélisation de surfaces	15
1.1 Introduction	15
1.2 Maillages polygonaux	18
1.2.1 Maillages <i>manifold</i>	18
1.2.2 Représentation d'un maillage polygonal	20
1.2.3 Surfaces de subdivision	21
1.3 Structures de données	23
1.3.1 Structure de données basée triangle	25
1.3.2 Structures de données basée arête	26
1.3.3 Structure de données basée demi-arête	27
1.4 Maillages triangulaires semi-réguliers et analyse multirésolution	28
1.4.1 Maillages triangulaires semi-réguliers	28
1.4.2 Analyse multirésolution	29
1.5 Critère de qualité utilisé	31
1.5.1 Distance point-surface	31
1.5.2 Distance unilatérale entre deux surfaces	31
1.5.3 Distance surface-surface	32
1.6 Conclusion	32
2 Contribution sur le remaillage : une approche basée sur la relaxation de Lloyd	35
2.1 Introduction	35
2.2 Etat de l'art	36

2.2.1	Remaillage semi-régulier basé paramétrisation	37
2.2.2	Remaillage semi-régulier ne se basant pas sur une paramétrisation	39
2.3	Motivation et contribution	40
2.4	Méthode proposée	40
2.4.1	Diagramme de Voronoï	40
2.4.2	Diagramme de Voronoï centroïdal	42
2.4.3	Description de l'algorithme proposé	46
2.5	Résultats expérimentaux	52
2.5.1	Objets utilisés	52
2.5.2	Pré-traitement des maillages d'entrée	52
2.5.3	Impact du seuil de subdivision ϵ	53
2.5.4	Impact de la méthode de sélection des sites du maillage grossier	54
2.5.5	Résultats visuels	56
2.5.6	Comparaison avec l'état de l'art	56
2.5.7	Discussion	56
2.6	Conclusion	61
3	Optimisation de deux transformées en ondelettes pour maillages semi-réguliers	63
3.1	Introduction	63
3.2	Etat de l'art	64
3.2.1	Ondelettes pour les maillages semi-réguliers	64
3.2.2	Compression basé-ondelettes des maillages semi-réguliers	65
3.3	Motivation et contributions	66
3.4	Notions de base	68
3.4.1	Transformée en ondelettes pour les maillages semi-réguliers	68
3.4.2	Schéma <i>lifting</i>	69
3.5	Schéma <i>lifting</i> adaptatif basé Butterfly	70
3.5.1	Optimisation de l'étape de prédiction	70
3.5.2	Calcul du nouveau filtre de mise à jour	73
3.5.3	Validation de la méthode proposée	75
3.6	Schéma <i>lifting</i> adaptatif basé Loop	77
3.6.1	Optimisation de la prédiction et du premier filtre de mise à jour	79
3.6.2	Calcul du deuxième filtre de mise à jour	79
3.6.3	Validation de la méthode proposée	82
3.7	Application à la compression	84
3.7.1	Principe d'un codeur géométrique	84
3.7.2	Description des codeurs testés	85
3.7.3	Résultats pour le schéma basé Butterfly	86
3.7.4	Résultats pour le schéma basé Loop	91
3.8	Conclusion	93
	Conclusion	99

II Maillages dynamiques	101
Introduction	103
4 Modélisation et compression d'animations 3D	107
4.1 Introduction	107
4.2 Séquences de maillages surfaciques	108
4.2.1 Notions générales	108
4.2.2 Animation par images-clés	108
4.3 Etat de l'art	110
4.3.1 Compression basée segmentation du maillage de la séquence . .	110
4.3.2 Compression basée sur une analyse en composantes principales	111
4.3.3 Compression basée sur une prédiction spatio-temporelle locale .	112
4.3.4 Compression basée sur des techniques multirésolutions	112
4.3.5 Compression basée sur un remaillage semi-régulier	114
4.4 Motivations et contributions	114
5 Remaillage semi-régulier de séquences de maillages surfaciques	115
5.1 Introduction	115
5.2 Présentation de la méthode proposée	116
5.3 Détails de l'approche proposée	116
5.3.1 Segmentation basée mouvement	116
5.3.2 Recalage et calcul du maillage moyen	117
5.3.3 Remaillage du maillage moyen	119
5.3.4 Remaillage de la séquence	122
5.4 Résultats expérimentaux	123
5.5 Conclusion	123
6 Compression des séquences de maillages semi-réguliers	129
6.1 Introduction	129
6.2 Présentation de l'approche proposée	130
6.3 Décomposition spatio-temporelle	131
6.3.1 Transformée en ondelette temporelle	131
6.3.2 Transformée en ondelettes spatiale	132
6.4 Allocation binaire	133
6.4.1 Formulation de la distorsion totale	133
6.4.2 Allocation binaire basée modèles	135
6.5 Résultats expérimentaux	136
6.6 Conclusion	137
Conclusion	143
Conclusion générale	145

Bibliographie	147
Annexe A Méthode de Nelder-Mead	159
Table des figures	161
Liste des tableaux	164
Publications	165

Remerciements

Les travaux de recherche présentés dans ce mémoire de thèse de doctorat en Automatique, Traitement du Signal et des Images (ATSI) ont été effectués au sein de l'équipe Images du Pôle Signal, Images et Systèmes (SIS) du laboratoire d'Informatique, Signaux et Systèmes de Sophia-Antipolis (I3S-UMR6070-UNS/CNRS), à ce titre, je tiens à remercier son Directeur Monsieur Luc PRONZATO, Directeur de Recherche au CNRS.

Je tiens tout particulièrement à remercier chaleureusement mon directeur de thèse, Monsieur Marc ANTONINI, Directeur de Recherche au CNRS, pour les conseils et la confiance dont il m'a honoré.

Je tiens à assurer toute ma reconnaissance et toute ma gratitude à mon Co-encadrant Monsieur Frédéric PAYAN, Maître de conférence à l'Université de Nice-Sophia Antipolis, pour l'intérêt qu'il a toujours porté à mes travaux de recherche.

Je tiens à remercier Monsieur Mohamed DAOUDI, Professeur à l'Institut Telecom à Lille 1, pour avoir accepté, en sus de ses nombreuses charges, celle de présider le jury de cette thèse de doctorat. Qu'il trouve ici le témoignage de ma grande reconnaissance.

Je tiens à remercier Madame Amel BENAZZA, Professeur à l'Ecole Supérieure des Communications de Tunis (SUP'COM), pour avoir acceptée de rapporter mes travaux de recherche et d'être membre du jury.

Je ne saurais dire toute ma gratitude à Monsieur Titus ZAHARIA, Professeur à l'Institut Télécom SudParis, de l'honneur qu'il me fait en acceptant de rapporter mes travaux de recherche et d'être membre du jury.

Je voudrais également remercier Monsieur Basile SAUVAGE, Maître de conférence à l'université de Strasbourg, qui m'a fait l'honneur d'accepter d'examiner ma thèse.

Que tous mes collègues et amis de l'équipe Images trouvent ici un témoignage de mon amitié pour la solidarité et le soutien moral dont ils ont toujours fait preuve.

Introduction générale

Les progrès scientifiques et technologiques dans les domaines du traitement d'images, des télécommunications et de l'infographie au cours de cette dernière décennie ont contribué à l'émergence de nouveaux médias, en particulier des données numériques en trois dimensions (3D). Aujourd'hui, le traitement, la transmission et la visualisation des objets 3D font partie des fonctionnalités possibles sur Internet.

L'émergence de la 3D est aussi directement liée à l'émergence des technologies d'acquisition 3D. En effet, les récents progrès dans l'acquisition par scanner 3D et les technologies de rendu 3D ont stimulés la création d'un vaste nombre de modèles 3D pour plusieurs domaines d'applications. Parmi ces domaines nous citons :

- le cinéma, dans lequel la 3D est à la mode. En effet, une grande partie des films qui sortent aujourd'hui exploitent la 3D, soit en utilisant une vision stéréoscopique, soit en intégrant des personnages réels dans des environnements 3D virtuels, ou encore en étant entièrement constitués d'animations 3D (voir figure 1) ;



FIGURE 1 – Extrait du film *Toy Story 3D* [2].

- les jeux vidéo, où les cartes graphiques utilisées ne cessent de progresser depuis plusieurs années. Ces cartes, qui embarquent actuellement quelques giga-octets de mémoire, sont composées par plusieurs milliards de transistors. Ajoutons que la progression importante dans la technologie de conception des cartes graphiques a permis le développement de divers types de jeux vidéo plus réalistes (voir fi-

gure 2) ;



FIGURE 2 – Extrait du jeux vidéo *Crysis* [3].

- la médecine, dans laquelle l’application de l’imagerie 3D a trouvé un grand succès tout au long de ces dernières années (notamment en médecine chirurgicale). Ainsi, l’imagerie 3D permet d’améliorer considérablement les gestes du chirurgien, et ce, à l’aide de simulateurs chirurgicaux ou des projections 3D des organes du patient ; ce qui permet d’aider le chirurgien à prévoir l’opération (voir figure 3) ;

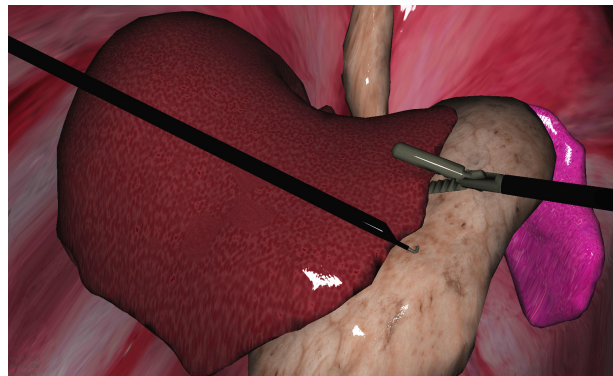


FIGURE 3 – Extrait d’un simulateur chirurgical [4].

- l’archéologie, qui nécessite l’utilisation des outils de conception assistée par ordinateur (CAO).

Cette activité croissante de la 3D a été rendue possible grâce au développement de matériels et logiciels pour les professionnels (en particulier la modélisation 3D et les outils de création et de manipulation) et pour le grand public (accélération graphique 3D matériel, Web3D, PDA de haute qualité, etc.). Dans ce contexte, de nouvelles générations de téléphones mobiles, qui permettent de visualiser des modèles 3D de façon

interactive, sont aussi proposées dans le marché.

Bien que la taille des disques durs et la vitesse de transmission des réseaux ne cessent d'augmenter, la compression reste une technique indispensable pour stocker, manipuler et transmettre plus aisément des données 3D, qui sont de plus en plus volumineuses. Cela est d'autant plus vrai pour les objets géométriques, comme les maillages surfaciques qui sont la représentation standard pour modéliser les objets géométriques 3D. Ces maillages sont composés à la fois d'une information géométrique (des sommets situés dans l'espace Euclidien traduisant la forme générale de la surface), mais aussi d'une information de connectivité (traduisant la manière dont sont connectés les sommets).

Ces maillages surfaciques représentent donc un outil puissant pour modéliser les objets 3D complexes car, ils permettent de modéliser n'importe quelle forme géométrique. Cependant, cela implique aussi une quantité importante d'informations à conserver pour une représentation fine et détaillée. En effet, il est nécessaire de conserver à la fois l'information géométrique et l'information de connectivité contrairement aux images fixes ou à la vidéo, où seules les valeurs des différents pixels sont nécessaires. Par conséquent, la compression des maillages surfaciques n'est pas aussi simple et évidente que dans les domaines du son, de l'image et de la vidéo.

La nécessité de conserver l'information de connectivité pour la reconstruction du maillage vient du fait que l'échantillonnage d'un maillage est irrégulier (tous les sommets n'ont pas forcément le même nombre de voisins) contrairement aux données précédemment citées, qui sont positionnées selon des grilles régulières. Ainsi, un maillage triangulaire doit être défini à l'aide d'une liste de sommets, qui sont décrits par leur position dans l'espace tridimensionnel Euclidien (3 nombres flottants), et une liste de triangles, qui sont décrits par les indices des sommets formant le triangle (3 entiers par triangle). Une représentation naïve et brute d'un maillage est donc extrêmement lourde, ce qui implique que la méthode de compression doit prendre en compte la complexité de cette représentation.

Une solution utilisée pour compresser efficacement les maillages est de réduire au maximum la redondance présente dans la connectivité. Cela s'effectue généralement à l'aide d'un parcours canonique le long des arêtes ou des triangles (encodage de la connectivité), suivi d'une quantification à précision fixe de la position des sommets (encodage de la géométrie), souvent à l'aide d'une technique de prédiction géométrique. Cette technique exploite la corrélation spatiale des sommets parcourus tout au long du codage de la connectivité afin de réduire l'entropie de la géométrie. Ces méthodes permettent un codage sans pertes puisque la connectivité est totalement conservée et que la géométrie ne subit aucune dégradation (à part la quantification à précision fixe). Cependant, cette approche permet d'atteindre des taux de compression peu élevés, étant donné que l'accent est mis sur l'optimisation du codage de la connectivité et qu'aucune perte n'est possible.

Cependant, il existe une autre alternative qui consiste à considérer le maillage ori-

ginal comme une instance parmi d'autres de la géométrie de la surface. Dans ce cas, nous parlons généralement de compression géométrique et non plus de compression de maillages. La compression géométrique considère que la géométrie est la composante la plus importante d'un maillage et que la connectivité peut, par conséquent, être modifiée pour optimiser l'encodage de la géométrie.

L'étape primordiale des méthodes de compression géométrique est l'étape de remaillage semi-régulier, qui modifie et réduit à son minimum l'information de connectivité nécessaire pour la reconstruction de l'objet. En plus, cette étape de remaillage permet d'obtenir des maillages surfaciques multirésolutions qui permettent de représenter facilement l'objet avec plusieurs niveaux de détails. Cette représentation multiéchelle a plusieurs avantages. Un de ces avantages est de permettre un codage, une transmission et un décodage progressif des maillages. En effet, une réduction de l'information à transmettre n'est pas le seul objectif d'une chaîne de compression optimale et le développement des réseaux de communications influe sur la mise au point de méthodes de compression progressive.

Le volume des maillages actuels et la bande passante limitée des réseaux encouragent le fait que nous puissions avoir très rapidement au décodage un aperçu grossier du maillage transmis. Ceci est rendu possible avec l'utilisation de maillages semi-réguliers multirésolutions. Ils permettent de transmettre tout d'abord le maillage de base à moindre coût, puis de transmettre les données supplémentaires raffinant l'objet jusqu'à obtention du maillage de plus haute résolution. La grande diversité des réseaux actuels et des supports pouvant envoyer ou recevoir des données multimédia donne aux méthodes progressives un avantage indéniable face aux méthodes monorésolutions.

De plus, en rendant l'échantillonnage "quasi-régulier", nous pouvons appliquer des outils performants d'analyse multirésolution, telle que la transformée en ondelettes, qui est fréquemment utilisée pour le traitement des images ou de la vidéo. Cela permet d'obtenir des méthodes de compression progressive et multirésolution de maillages surfaciques très performantes.

L'idée basique de ces méthodes de compression, qui sont basées sur l'analyse multirésolution de maillages semi-réguliers, est d'optimiser le compromis entre la taille de la trame binaire et la qualité visuelle du maillage reconstruit. Ceci est effectué en contrôlant et en minimisant les pertes dues à la quantification de la géométrie en fonction d'un certain débit ou d'une certaine taille de la trame binaire. Ce principe, appelé allocation binaire, est un outil essentiel pour optimiser les codeurs exploitant l'analyse multirésolution. L'idée est de quantifier finement la géométrie (les coefficients d'ondelettes) qui apporte des détails visuels et de quantifier grossièrement, voire même négliger, la géométrie qui apporte peu de détails. Bien que l'allocation binaire soit fréquemment utilisée pour la compression de l'image ou de la vidéo, elle est encore peu présente parmi les méthodes de compression de maillages triangulaires.

L'objectif visé, dans cette thèse de doctorat, est donc de proposer des solutions face aux problèmes de compression et d'affichage avec plusieurs niveaux de détails des objets

3D. Les travaux développés sont répartis en deux parties. La première partie est réservée à l'étude des maillages statiques, où nous détaillons les solutions proposées pour les objets 3D surfaciques. Quant à la deuxième partie, elle concerne l'étude des maillages dynamiques, où nous détaillons les solutions proposées pour les objets 3D animés.

Première partie

Maillages statiques

Introduction

En théorie de l'information, la compression avec pertes est une technique qui réduit fortement la quantité d'information à stocker pour représenter des données numériques. En effet, cette technique permet d'éliminer une partie des informations initiales (les moins significatives) produites par la source. La procédure vise à minimiser la quantité de données qui doit être stockée, manipulée et/ou transmise sur les réseaux par un ordinateur.

L'analyse multirésolution et la transformée en ondelettes sont des techniques essentielles dans un système de compression avec pertes. En effet, la transformée en ondelettes permet de décomposer un signal numérique en un signal basse résolution et des ensembles (sous-bandes) de détails fins, appelés coefficients d'ondelettes. Dans la pratique, beaucoup de coefficients d'ondelettes sont proches de zéro. L'avantage du processus de décomposition est que des bonnes approximations du signal à compresser peuvent être obtenues, même si certains des coefficients sont mis à zéro. Une telle décomposition est importante pour une compression efficace, puisque, d'une part elle permet de représenter un signal selon une structure multi-échelle (une version simplifiée du signal qui pourra être affinée par la suite en ajoutant des sous-bandes de coefficients), et d'autre part elle permet la dé-corrélation du signal en une partie basse fréquence à forte énergie et des sous-bandes haute fréquence à faible variance. Ces deux propriétés permettront au final une représentation plus compacte de la surface [109].

Avant les travaux pionniers de Lounsbery *et al.* [97] dans le domaine des maillages surfaciques, les transformées en ondelettes étaient utilisées uniquement sur des signaux définis sur des grilles d'échantillonnage régulières, comme par exemple, les données audio, les images et les vidéos. En effet, Lounsbery *et al.* [97] ont étendu la transformée en ondelettes aux surfaces *manifold* (voir section 1.2.1) de topologie arbitraire.

En général, il est impossible de garder une régularité de l'échantillonnage pour les surfaces *manifold*, à l'inverse des données audio, des images et des vidéos, car elles sont généralement représentées par des maillages avec une connectivité irrégulière [132]. Cependant, en créant des maillages semi-réguliers, une extension de la transformée en ondelettes cartésienne pour les maillages *manifold* devient alors possible [127].

Les maillages triangulaires semi-réguliers sont construits par un processus de subdi-

vision quaternaire récursive des triangles. A partir d'un maillage de base grossier, dont les sommets sont des échantillons de la surface *manifold* à approximer, chaque triangle est divisé en quatre sous-triangles. Les positions des nouveaux sommets ajoutés représentent à nouveau des échantillons de la même surface. Ce processus est répété récursivement, en produisant une hiérarchie de maillages multirésolution jusqu'à obtenir un maillage qui approxime la surface *manifold*, qui est donnée avec une précision suffisante. Ainsi, le maillage semi-régulier obtenu possède un ensemble de *patches* triangulaires ayant une connectivité régulière.

Une transformée en ondelettes est appliquée itérativement entre deux maillages voisins dans la hiérarchie de maillages multirésolutions, qui représente le maillage semi-régulier (i.e., entre un maillage grossier et un maillage plus fin). Cette transformée prédit les positions des sommets du maillage le plus fin, qui sont ajoutés lors de la subdivision quaternaire des triangles, et ce, à partir des positions des sommets du maillage le plus grossier. La prédiction est habituellement réalisée en utilisant des techniques de subdivision de surfaces [144]. Les erreurs de prédiction représentent les détails ou les coefficients d'ondelettes. Ces détails sont analogues aux détails obtenus dans un cadre traditionnel de traitement du signal, par exemple, les détails obtenus après un filtrage passe-haut d'une image.

Ainsi, un schéma de compression géométrique pour les maillages se compose généralement de trois étapes (voir figure 1), à savoir :

1. **remaillage semi-régulier**, qui transforme un maillage *manifold* donné en un maillage semi-régulier ;
2. **transformée en ondelettes**, qui décompose le maillage semi-régulier en un maillage grossier et des ensembles de détails ;
3. **codage géométrique**, qui comporte le codage des différentes sous-bandes basse fréquence et hautes fréquences ainsi que le codage de la connectivité du maillage grossier.

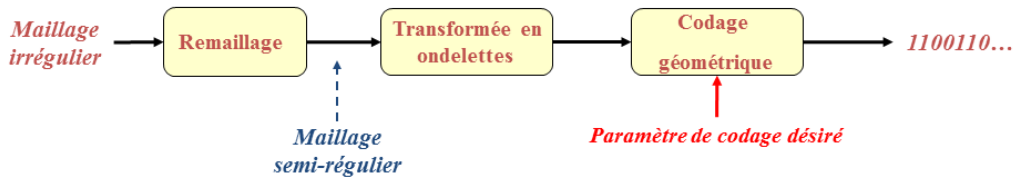


FIGURE 1 – Schéma général de codage progressif pour les maillages surfaciques.

Ainsi, motivés par l'amélioration des différents étages de la compression basée ondelettes des maillages surfaciques triangulaires, nous présentons, dans cette partie, une nouvelle approche pour le remaillage semi-régulier et une technique d'optimisation pour deux transformées en ondelettes pour les maillages surfaciques semi-réguliers.

Cette première partie est divisée en trois chapitres. Dans le premier chapitre, nous introduisons les notions sur les surfaces et les maillages surfaciques. Dans le deuxième chapitre, nous détaillons la technique proposée pour le remaillage semi-régulier des maillages surfaciques, qui sont nécessaires pour la bonne compréhension du reste de cette partie. Dans le troisième et dernier chapitre, nous détaillons notre contribution, qui permet d'améliorer les techniques d'analyse multirésolution des maillages semi-réguliers et, par conséquent, l'étage de codage positionné en aval de la chaîne de compression.

Chapitre 1

Modélisation de surfaces

Dans ce chapitre, nous présentons les différents types de représentations d'un objet géométrique. Etant donné que cette représentation peut être bi-dimensionnelle (2D) ou tri-dimensionnelle (3D), nous présentons uniquement les surfaces 2D qui délimitent un objet 3D. Nous commençons par introduire, dans la section 1.1, les différents types de représentations de surfaces. Ensuite, nous introduisons, dans la section 1.2, les maillages polygonaux. Par la suite, nous donnons, dans la section 1.3, quelques structures de données, qui sont utilisées pour décrire les maillages triangulaires et notamment celle que nous avons choisi d'utiliser durant nos travaux. Ensuite, dans la section 1.4, nous présentons les particularités des maillages semi-réguliers et le principe de l'analyse multirésolution. Enfin, dans la section 1.5, nous présentons le critère de qualité que nous avons utilisé dans nos résultats expérimentaux.

1.1 Introduction

Actuellement, les objets 3D sont plus complexes à gérer que d'autres données multimédia, tels que les signaux audio et les images 2D. En effet, il existe différentes représentations pour de tels objets.

Par exemple, une image 2D a une représentation unique et assez simple : une grille 2D ($n \times n$) est composée par n^2 éléments (*pixels*) représentant une valeur de couleur ou un niveau de gris. Ce type de représentation, qui est standardisé, est utilisé par les différents dispositifs et techniques qui produisent des images numériques (caméras numériques, scanners, etc.).

Pour les modèles 3D, il y a différents types de représentations : un objet peut être représenté sur une grille 3D ou dans l'espace Euclidien 3D. Dans ce dernier cas, l'objet peut être exprimé à l'aide d'une équation mathématique (par exemple, une surface implicite), d'un ensemble de facettes approximant la surface du modèle original, ou encore à l'aide d'un ensemble de surfaces mathématiques.

Les principales difficultés dans le domaine de la 3D sont les suivantes :

- les différentes sources de données 3D (tomographie, balayage laser, etc.) ne produisent pas les mêmes types de représentations ;
- les différentes applications (conception assistée par ordinateur, médicale, etc.) n'utilisent pas les mêmes types de représentations ;
- la conversion d'un type de représentation à un autre est assez complexe et constitue souvent un problème ouvert.

En pratique, un scanner laser 3D produit un ensemble de points dans l'espace 3D, où chaque point est défini par ses coordonnées x , y et z dans l'espace Euclidien. La figure 1.1.a montre un nuage de points représentant l'objet BUNNY. Cette représentation, qui est à la base de points fournis par des scanners, n'est pas efficace pour le calcul des propriétés physiques et géométriques ou à l'affichage sur un écran. En effet, il n'existe pas de relations de voisinage entre les points 3D et il n'y a aucune information surfacique ou volumique.

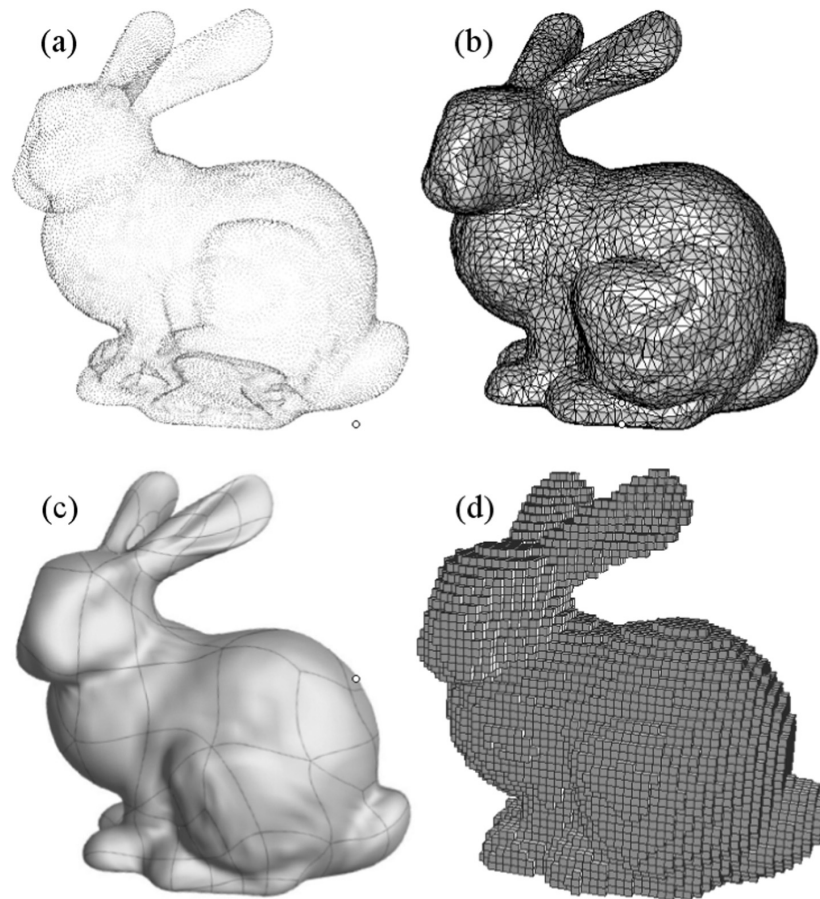


FIGURE 1.1 – Différentes représentations de BUNNY : (a) un nuage de points, (b) un maillage triangulaire, (c) un ensemble de surfaces paramétriques, (d) un ensemble de voxels (images extraites de [44]).

Par conséquent, ces représentations sont souvent converties en maillages polygonaux, et plus particulièrement en maillages triangulaires (voir la figure 1.1.b). Avec cette représentation polygonale, l'objet est représenté par la surface qui le délimite. Cette surface est composée d'un ensemble de points 3D (les sommets), qui sont reliés par des arêtes pour former un ensemble de facettes polygonales. De nombreuses techniques de reconstruction de surface existent pour produire un maillage triangulaire à partir d'un nuage de points, qui est issu d'un scanner 3D [17, 82].

Les maillages polygonaux peuvent représenter des surfaces ouvertes ou fermées et de topologie arbitraire, avec une précision qui dépend du nombre de sommets et de facettes. Les algorithmes d'intersection, de détection, de collision ou de rendu sont simples et rapides avec ce type de représentation, car la manipulation de facettes planes (en particulier des triangles) est aussi simple (algèbre linéaire). Cette rapidité est particulièrement utile pour les jeux vidéo. Ces avantages font de ce modèle la représentation la plus répandue pour les objets 3D [44].

Toutefois, les maillages polygonaux ont certaines limites. Ce type de représentation est intrinsèquement discret, puisque le nombre de sommets et de facettes dépend de la précision attendue, donc une précision élevée peut conduire à une énorme quantité de données. Par ailleurs, la définition de la forme est très locale et, par conséquent, l'action d'appliquer une déformation globale ou de créer manuellement facette par facette un maillage polygonal est assez difficile.

La nécessité d'une représentation 3D adaptée à la modélisation et à la conception a conduit à l'apparition de surfaces paramétriques. Cette famille de surfaces 3D (comme les surfaces de Bézier, les B-splines et les surfaces NURBS) est particulièrement utilisée pour la conception assistée par ordinateur (CAO), la fabrication assistée par ordinateur (FAO) et l'ingénierie assistée par ordinateur (IAO). Ces surfaces paramétriques permettent d'obtenir une représentation exacte contrairement aux maillages polygonaux, qui ne représentent qu'une approximation. Étant donné que ces surfaces sont définies sur un domaine paramétrique, alors elles ne peuvent pas représenter facilement une forme avec une topologie arbitraire et, par conséquent, l'objet est souvent représenté par un *patchwork* de surfaces paramétriques. Par exemple, l'objet BUNNY de la figure 1.1.c est modélisé avec 153 *patches* de *B-spline* bicubiques.

Ces surfaces ont des propriétés tangentielles et normales (courbures) continues, qui les rendent très utiles pour la conception. Par ailleurs, elles sont mathématiquement complètes et permettent la modélisation d'une grande variété de formes. Ces surfaces ne sont définies que par quelques ensembles de points de contrôle, au lieu d'un ensemble dense de sommets, ce qui les rend beaucoup plus compactes que les maillages polygonaux. Leur principal inconvénient est qu'elles sont plus complexes à manipuler qu'un ensemble de triangles.

Les maillages polygonaux et les surfaces paramétriques sont des représentations par les bords (B-Rep), car ils ne modélisent un objet que par la surface qui le délimite. Plusieurs domaines (en particulier l'imagerie médicale) ont besoin des données intérieures

d'un objet 3D et, par conséquent, ils utilisent un modèle volumique discret. Ce modèle ne représente pas l'objet 3D dans l'espace Euclidien, mais dans une grille 3D similaire à la représentation d'images 2D. Chaque élément de la grille est un *voxel* (abréviation de *pixel* volumique). Ainsi, l'objet est représenté par l'ensemble des *voxels* constituant son volume. Un *voxel* peut contenir une valeur booléenne (à l'intérieur ou à l'extérieur de l'objet) ou d'autres informations, comme les densités locales. Les appareils médicaux (par exemple, IRM) produisent souvent ces données volumiques pour décrire l'intérieur d'un organe. La figure 1.1.d montre une représentation voxelique de l'objet BUNNY, et ce, dans une grille de 50×50 *voxels*.

1.2 Maillages polygonaux

Un maillage surfacique polygonal est défini par un ensemble de polygones (voir figure 1.2). Ainsi, ce type de maillage contient trois types d'éléments, à savoir : sommets, arêtes et facettes (les plus utilisés sont les triangles). Des attributs supplémentaires peuvent être joints aux sommets, comme les vecteurs normaux, la couleur et la texture. Un maillage polygonal se compose de deux types d'informations. Il s'agit de la géométrie et de la connectivité. La géométrie décrit la position des sommets dans l'espace 3D et la connectivité décrit comment sont connectés ces sommets, à savoir : les différentes relations qui existent entre les éléments du maillage. Ces relations précisent, pour chaque facette, les bords et les sommets qui la composent, et pour chaque sommet, les arêtes et les facettes incidentes. La valence d'un sommet est le nombre de ses arêtes incidentes et le degré d'une facette est son nombre d'arêtes (voir figure 1.2).

Soulignons que n'importe quel objet complexe de topologie arbitraire peut être modélisé par un maillage polygonal, pourvu qu'il y ait suffisamment d'espace mémoire disponible. Enfin, il existe de nombreuses techniques de génération de maillages polygonaux à partir d'autres modèles géométriques (surfaces paramétriques, surfaces implicites, modèles discrets, etc.) [16, 96]. Ce type de représentation domine largement les différents formats 3D, tels que : VRML et MPEG-4 [44].

Néanmoins, ce modèle a plusieurs limites ; il offre seulement une représentation discrète. En effet, un maillage polygonal possède seulement une continuité C^0 . Par suite, il ne peut pas représenter exactement une surface donnée et représente seulement une approximation, dont la précision (liée au nombre de polygones) dépend de l'application ciblée (rendu, détection de collision, etc.). Par conséquent, ce type de représentation peut devenir assez lourd, en termes de quantité de données, si la précision demandée est élevée ou si l'objet à représenter est complexe.

1.2.1 Maillages *manifold*

Un maillage polygonal est appelé *manifold* si chacune de ses arêtes appartient à une (dans le cas d'un bord) ou à deux facettes. Un maillage *manifold* est associé à sa

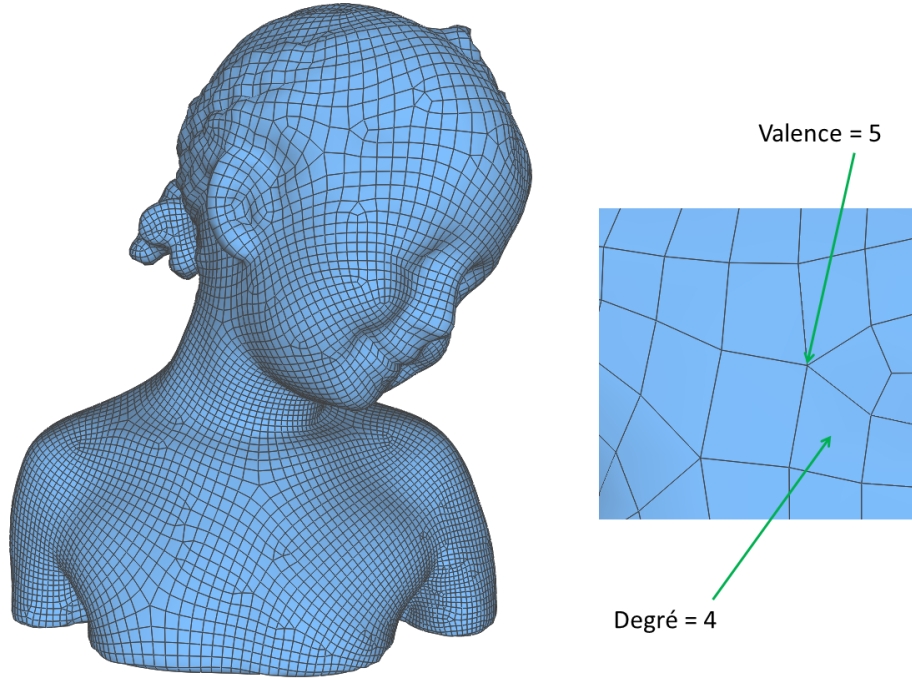


FIGURE 1.2 – Exemple de maillage polygonal (15653 sommets, 15770 facettes, 31421 arêtes) illustrant la valence d'un sommet et le degré d'une facette.

caractéristique d'Euler-Poincaré χ , soit :

$$\chi = v - e + f, \quad (1.1)$$

avec v , e et f les nombres de sommets, arêtes et facettes du maillage, respectivement.

La caractéristique d'Euler-Poincaré est liée au genre de la surface correspondante, et ce, selon l'équation suivante :

$$g = \frac{2c - b - \chi}{2}, \quad (1.2)$$

où c est le nombre de composantes connexes et b représente le nombre de bords.

Le genre d'une surface décrit sa complexité topologique. En effet, il représente le nombre maximum de courbes fermées simples sans points communs pouvant être tracé à l'intérieur de cette surface sans la déconnecter. Autrement dit, dans le procédé de détermination du genre, le complément de la réunion de ces courbes reste connexe. Plus concrètement, si nous considérons que la surface est en papier, alors le genre est le nombre maximal de découpages fermés faisables, sans que la surface ne soit séparée en plusieurs morceaux. C'est une notion de topologie, dans laquelle deux surfaces n'ayant pas le même genre ne sont pas homéomorphes. Par exemple, une sphère et un disque ont un genre de 0, alors qu'un tore a un genre de 1. La figure 1.3 montre le genre de quelques surfaces.

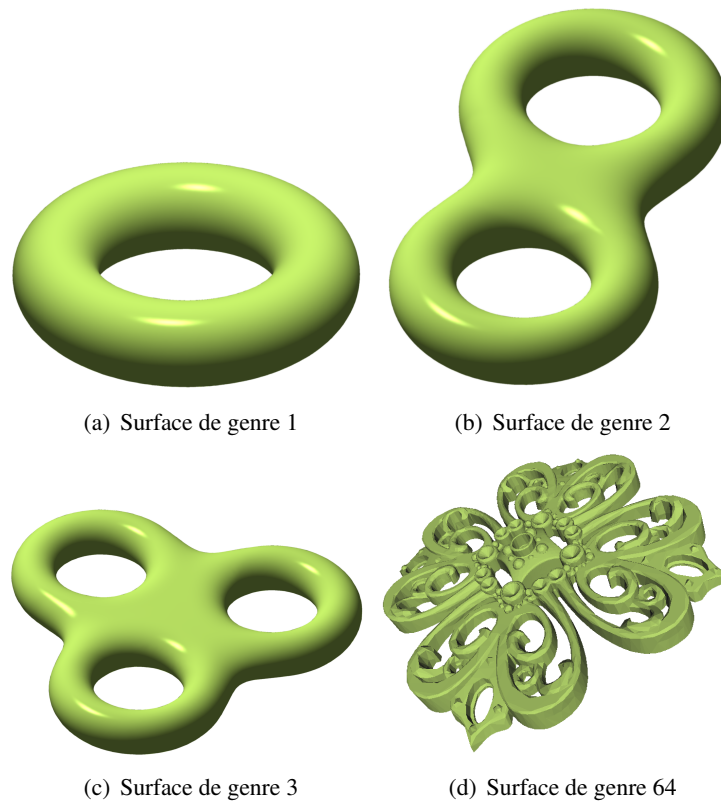


FIGURE 1.3 – Genre de quelques surfaces.

1.2.2 Représentation d'un maillage polygonal

Le format standard (utilisé, par exemple, dans le standard VRML) permettant de décrire un maillage polygonal est défini comme suit :

- la géométrie est représentée par une liste de sommets, chacun d'eux étant représenté par ses coordonnées dans l'espace Euclidien 3D ;
- la connectivité est décrite par la liste des facettes, chacune d'elles étant représentée par une liste cyclique d'indices de ses sommets incidents.

La figure 1.4 montre cette représentation standard. Le fichier contient les coordonnées des sommets v_0, v_1, v_2, v_3, v_4 et v_5 (la géométrie) et, pour chaque facette F_0, F_1, F_2 et F_3 , les indices des sommets correspondants (la connectivité).

Bien que cette méthode de description des maillages polygonaux présente l'avantage d'être simple, les données codées sont très redondantes, puisque les sommets et les arêtes sont évoqués à plusieurs reprises. D'ailleurs, les méthodes de compression ont pour but d'éliminer cette redondance.

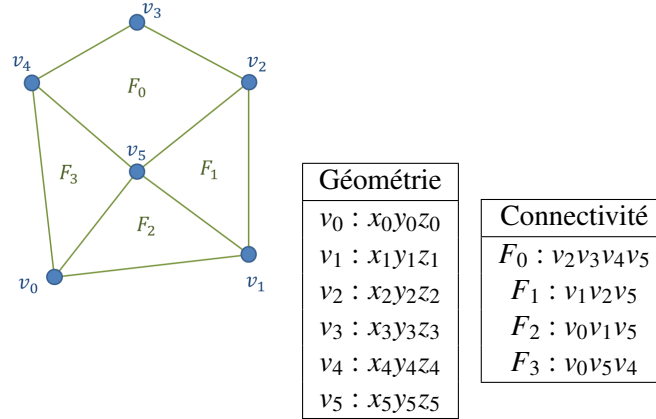


FIGURE 1.4 – Exemple d’une représentation standard pour un maillage simple contenant six sommets et quatre facettes.

1.2.3 Surfaces de subdivision

Le but de la subdivision est de définir une forme lisse à partir d’un polyèdre grossier en ajoutant des sommets et des arêtes d’une manière itérative, tout en respectant certaines règles de subdivision. La figure 1.5 illustre ce mécanisme de subdivision pour une courbe 2D. La figure 1.5.a montre quatre points reliés par des segments (le polygone de contrôle), tandis que la figure 1.5.b montre la courbe polygonale après une étape de raffinement (trois points ont été ajoutés et les anciens points ont été déplacés). En répétant ce processus de raffinement plusieurs fois, la courbe polygonale tend vers une courbe limite lisse (voir figure 1.5.d). Les propriétés de continuité de la surface limite dépendent des règles de subdivision. De même, un exemple de surface de subdivision est donné figure 1.6. A chaque itération, chaque quadrangle est divisé en quatre. Des nouveaux sommets sont donc insérés et les anciens sont déplacés. Après quelques itérations de raffinement, la surface semble lisse (voir figure 1.6.d).

La principale difficulté est de trouver des règles de subdivision qui conduisent à de bonnes propriétés, telles que : la simplicité de calcul, la continuité et le bon aspect visuel. Dans le cas de la courbe de subdivision présentée figure 1.5, la courbe polygonale est d’abord subdivisée linéairement (à savoir, les sommets sont ajoutés dans le milieu des segments), et ensuite chaque sommet v_i est remplacé par une combinaison linéaire de lui-même et de ses voisins directs v_{i-1} et v_{i+1} , en appliquant le masque de subdivision suivant :

$$v'_i = \frac{1}{4}(v_{i-1} + 2v_i + v_{i+1}). \quad (1.3)$$

Avec ces règles, la courbe limite correspond à une B-spline cubique (voir figure 1.5).

En 1978, Doo et Sabin [41] et Catmull et Clark [25] ont été les premiers à introduire des règles de subdivision pour les surfaces. Leurs schémas ont généralisé les *B-splines* biquadratiques et bicubiques [49]. Aujourd’hui, de nombreux schémas de subdivision sont développés. Nous pouvons distinguer les schémas basés sur des maillages quadran-

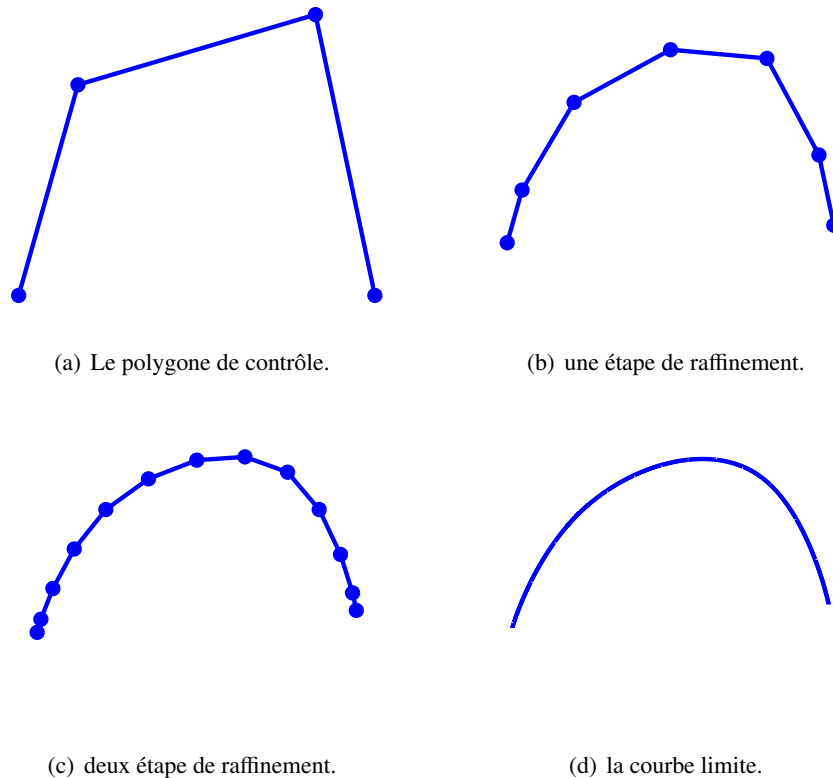


FIGURE 1.5 – Exemple d’une courbe de subdivision.

gulaires [81, 69] et les schémas basés sur des maillages triangulaires [94, 145]. Récemment, Stam et Loop [125], et Schaefer et Warren [120] ont introduit des schémas de subdivision qui peuvent opérer à la fois sur les maillages quadrangulaires et les maillages triangulaires. Par ailleurs, en 1994, Hoppe *et al.* ont introduit des règles de subdivision spéciales pour gérer des arêtes vives, c’est-à-dire de conserver la netteté des bords de la surface [66].

Un schéma de subdivision peut être décrit par :

- un aspect topologique, où chaque schéma de subdivision modifie la connectivité d’un maillage polygonal, en ajoutant et supprimant des sommets ou en effectuant des *flips* d’arêtes. Nous pouvons distinguer deux principales catégories de schémas de subdivision. Il s’agit des schémas primaux, qui ne suppriment pas les anciens points, et des schémas duaux, qui permettent de les supprimer ;
- un aspect géométrique, où la modification des positions des sommets peut être considérée comme une opération de lissage sur le maillage en entrée. Nous distinguons deux catégories principales, à savoir : les schémas interpolants, qui ne modifient pas la position des anciens sommets, et les schémas non-interpolants, qui, au contraire, permettent de les changer.

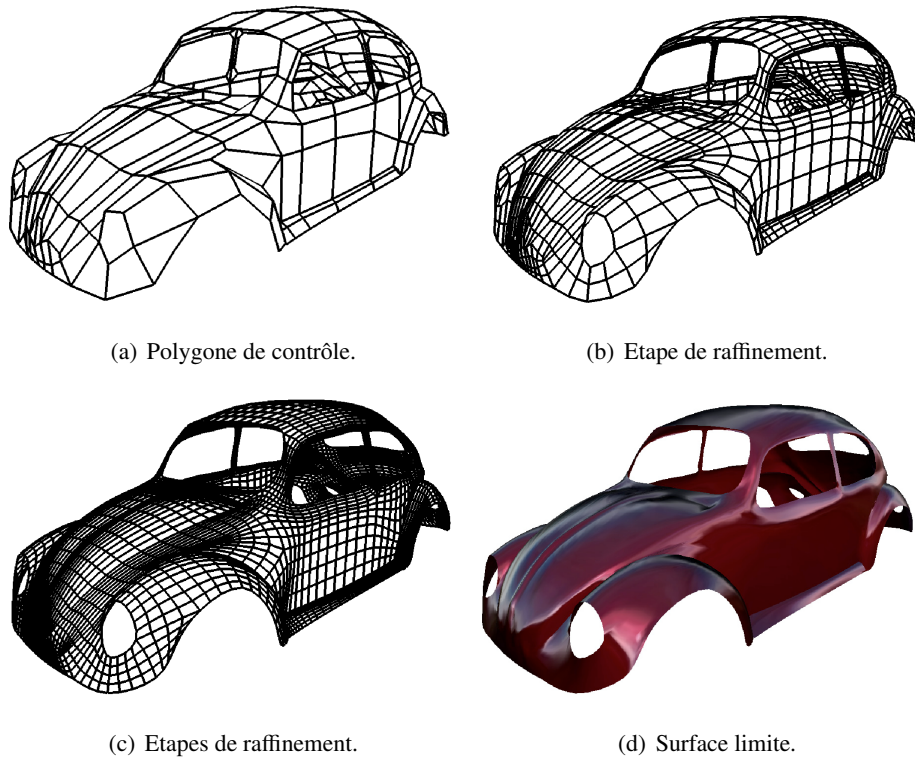


FIGURE 1.6 – Exemple d’une surface de subdivision (images tirées de [19]).

Le tableau 1.1 donne une classification des principaux schémas de subdivision, qui sont publiés dans la littérature.

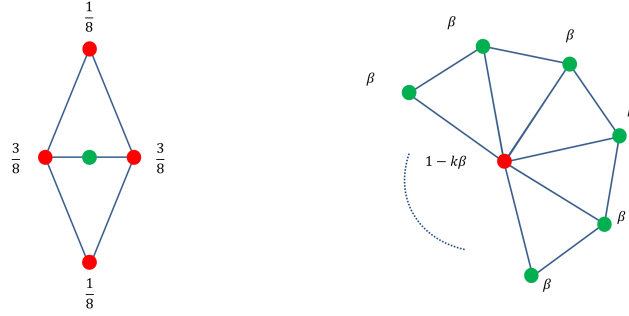
TABLE 1.1 – Classification des principaux schémas de subdivision [44].

	Primal (Triangles)	Primal(Quadrangles)	Dual (Quadrangles)
Non-interpolant	Loop (C^2) [94]	Catmull–Clark (C^2) [25]	Doo–Sabin (C^1) [41] Midedge (C^1) [69]
Interpolant	Butterfly (C^1) [145]	Kobbelt (C^1) [81]	-

Généralement, il est possible de décrire un schéma de subdivision par un ensemble de masques. Par exemple, la figure 1.7 donne les différents masques utilisés dans le schéma de subdivision de Loop [94], qui s’appliquent sur les maillages triangulaires.

1.3 Structures de données

La façon la plus simple pour représenter un maillage triangulaire consiste à représenter la géométrie par une liste de coordonnées des sommets et la topologie par une liste de triangles (chaque triangle est représenté par les indices de ses sommets incidents), comme nous l’avons décrit dans la sous-section 1.2.2. Avec une telle représentation,



(a) Masque pour un nouveau sommet qui n'est pas sur un bord. (b) Masque pour un ancien sommet qui n'est pas sur un bord.



(c) Masque pour un nouveau sommet qui est sur un bord. (d) Masque pour un ancien sommet qui est sur un bord.

FIGURE 1.7 – Différents masques utilisés dans le schéma de subdivision Loop. $\beta = \frac{1}{8}(\frac{5}{8} - (\frac{3}{8} + \frac{1}{4} \cos(\frac{2\pi}{n}))^2)$, avec n la valence du sommet [94].

chacune des trois coordonnées de chaque sommet est codée avec 4 octets. Pour chaque face, chacun des trois indices de sommets est codé avec 4 octets. Ainsi, la taille du fichier qui décrit le maillage est égal à $3 \times 4 \times (N_v + N_f)$, avec N_v le nombre de sommets et N_f le nombre de triangles. Par conséquent, pour un maillage régulier, où $NF \approx 2 \times N_v$, il faut prévoir $3 \times 4 \times 3 = 36$ octets (288 bits) par sommet.

Cette représentation n'est pas optimale pour la plupart des applications. Par exemple, les informations de voisinage ne sont pas explicites et les données codées sont très redondantes, puisque les sommets et les arêtes sont évoqués à plusieurs reprises. Il en résulte que cette structure de données nécessite des recherches coûteuses pour récupérer les informations d'adjacence locale d'un sommet. Par conséquent, elle n'est pas assez efficace pour la plupart des algorithmes.

Les opérations sur les maillages triangulaires les plus fréquemment utilisés par les algorithmes sont les suivantes [20] :

- accéder aux différents sommets, arêtes et facettes ;
- trouver l'arête suivante (ou précédente) d'une facette, en partant d'une arête donnée ;
- accéder aux facettes incidentes à une arête donnée ;
- accéder aux sommets d'une arête ;
- accéder à au moins une arête ou à une facette incidente ayant un sommet donné.

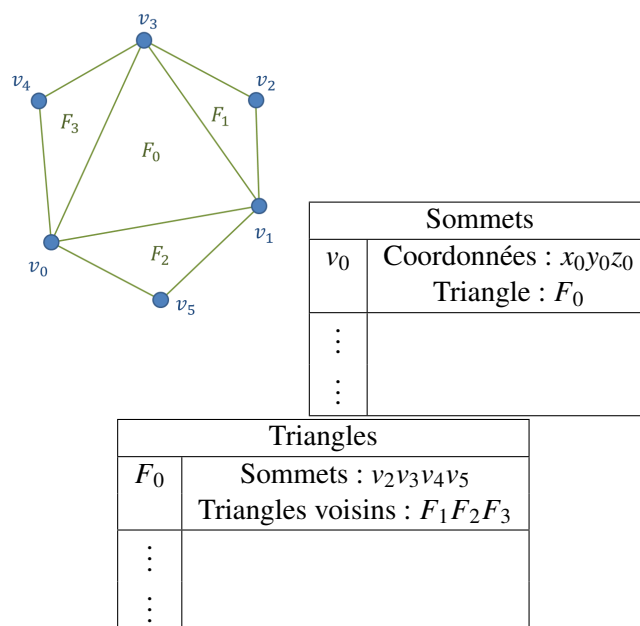


FIGURE 1.8 – Exemple d’une représentation basée triangle.

Par la suite, nous donnons une description des différentes structures de données, qui permettent de faciliter les opérations citées ci-dessus. Pour nos implémentations, nous avons utilisé la structure de données basée demi-arête (voir la sous-section 1.3.2). En effet, comme nous allons voir dans les sous-sections qui suivent, cette structure de données permet d’énumérer facilement les voisins d’un sommet donné. Ce calcul de voisinage est souvent utilisé dans nos implémentations et, par conséquent, cette structure de données nous permet de réduire le temps de calcul global.

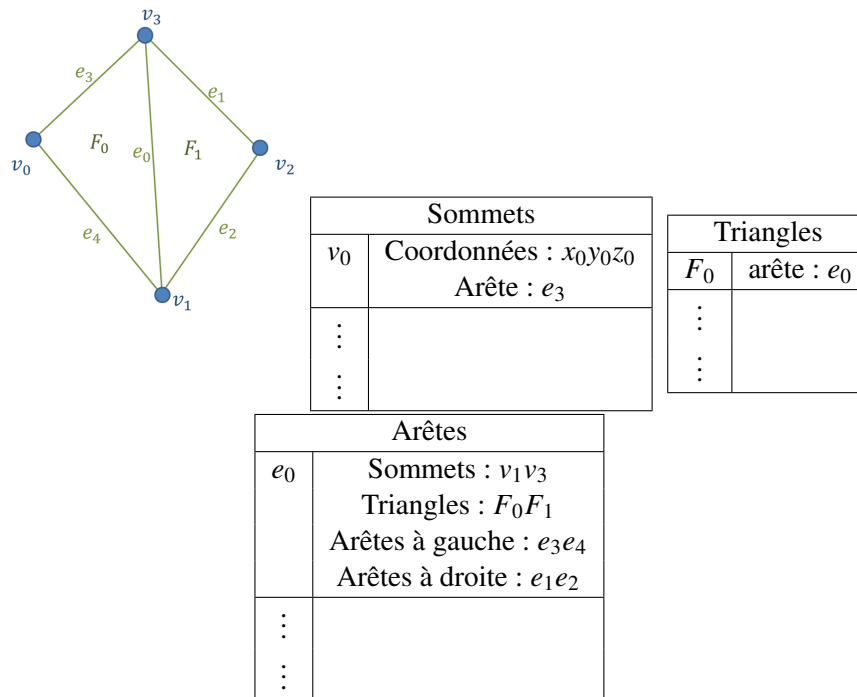
1.3.1 Structure de données basée triangle

Une structure de données basé triangle (voir figure 1.8) consiste à stocker :

- pour chaque triangle du maillage, les trois sommets qui le composent, ainsi que les références des triangles qui lui sont voisins ;
- pour chaque sommet du maillage, la référence d’un triangle incident, ainsi que ses coordonnées 3D.

Avec cette structure de données, il est possible d’énumérer tous les triangles incidents à un sommet donné, ainsi que d’accéder directement aux voisinages d’un triangle donné. Pour un maillage triangulaire *manifold*, il faut compter $3 \times 4 + 1 \times 4 = 16$ octets par sommet et $3 \times 4 + 3 \times 4 = 24$ octets par triangle. Compte-tenu de la formule d’Euler-Poincaré (équation 1.1), le nombre de triangles est à peu près égal à deux fois le nombre de sommets. Par conséquent, il faut prévoir 64 octets (512 bits) par sommet.

Cependant, cette structure de données a aussi quelques inconvénients, à savoir :

FIGURE 1.9 – Exemple d’une représentation basée arête (structure *winged-edge*).

- les arêtes ne sont pas stockées explicitement. Par exemple, aucune donnée ne peut être attachée à une arête donnée ;
- l’énumération du voisinage d’un sommet donné nécessite de chercher sa position dans un triangle qui lui est incident.

1.3.2 Structures de données basée arête

Il existe plusieurs structures de données basée arête, les plus connues étant les structures *winged-edge* [13] et *quad-edge* [58]. Cette dernière se décline en plusieurs variantes, comme celle proposée dans [108]. La structure *winged-edge*, telle que donnée figure 1.9, présente les propriétés suivantes :

- pour chaque sommet, nous stockons ses coordonnées et une référence d’une arête incidente ;
- pour chaque triangle, nous stockons une référence d’une arête incidente ;
- pour chaque arête, nous stockons ses sommets, les références de ses triangles incidents, ainsi que les références des arêtes se trouvant à gauche et à droite.

Pour un maillage triangulaire *manifold*, il faut compter $(3 \times 4 + 1 \times 4 = 16)$ octets par sommet, 4 octets par triangle et $(8 \times 4 = 32)$ octets par arête. En se basant sur l’équation 1.1 d’Euler-Poincaré, nous pouvons affirmer que le nombre de triangles est à peu près égal à deux fois le nombre de sommets et le nombre d’arêtes est à peu près égal au triple du nombre de sommets. Par conséquent, il faut prévoir 120 octets (960 bits) par sommet.

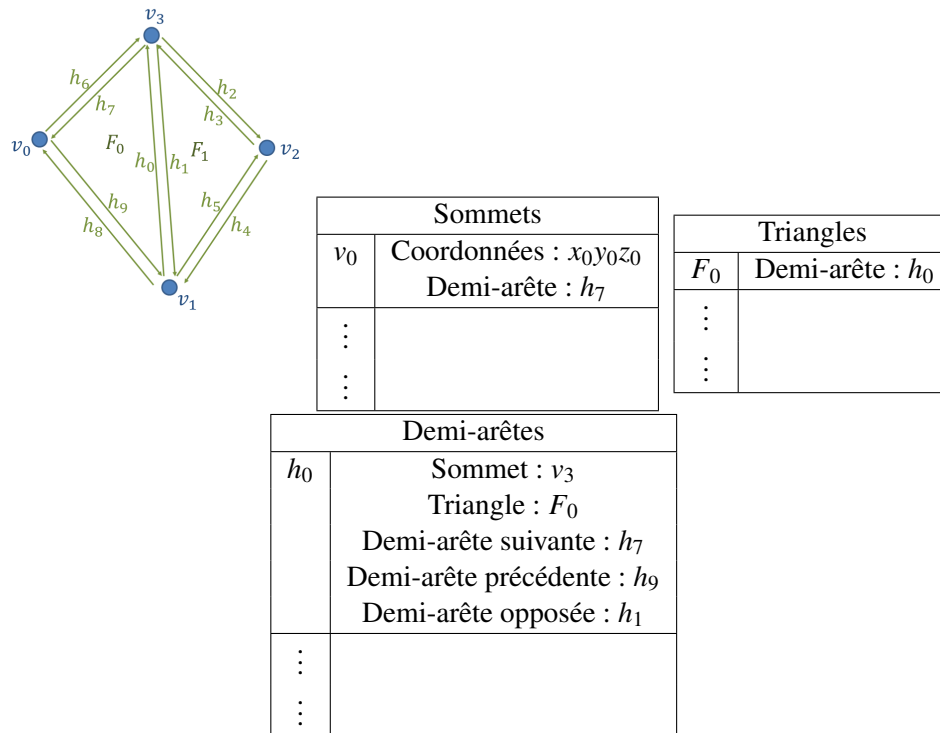


FIGURE 1.10 – Exemple d’une représentation basée demi-arête.

L’inconvénient majeur de cette structure est qu’il faut chercher la position d’un sommet donné dans une de ses arêtes incidentes afin d’énumérer son voisinage. Cet inconvénient est néanmoins résolu grâce à la structure de données basée demi-arête, que nous allons décrire dans la sous-section suivante.

1.3.3 Structure de données basée demi-arête

La structure de données basée demi-arête [76, 105] se distingue par rapport aux autres structures de données par le fait qu’elle divise toute arête en deux demi-arêtes orientées. Cette structure de données est capable de représenter des maillages triangulaires arbitraires et *manifold*. Dans cette structure de données, les demi-arêtes sont orientées dans le sens trigonométrique autour de chaque triangle (voir figure 1.10). Cette structure présente les propriétés suivantes :

- pour chaque sommet, nous stockons ses coordonnées, ainsi qu’une référence vers une demi-arête incidente ;
- pour chaque triangle, nous stockons une demi-arête incidente ;
- pour chaque demi-arête, nous stockons une référence du sommet vers lequel elle pointe, trois références vers les demi-arêtes qui les précède, qui les suit et qui lui sont opposées, ainsi qu’une référence vers le triangle incident.

En utilisant cette structure de données, et pour un maillage triangulaire *manifold*, il faut compter ($3 \times 4 + 1 \times 4 = 16$) octets par sommet, 4 octets par triangle et ($5 \times 4 = 20$) octets par demi-arête (soit 40 octets par arête). Compte-tenu de l'équation 1.1 d'Euler-Poincaré, nous pouvons dire que le nombre de triangles est à peu près égal à deux fois le nombre de sommets et le nombre d'arêtes est à peu près égal au triple du nombre de sommets. Par conséquent, il faut prévoir 144 octets (1152 bits) par sommet.

1.4 Maillages triangulaires semi-réguliers et analyse multirésolution

1.4.1 Maillages triangulaires semi-réguliers

Les maillages triangulaires semi-réguliers, qui sont des maillages de subdivision, correspondent donc aux maillages multirésolutions. Ces maillages sont aujourd'hui très répandus pour représenter les surfaces. La figure 1.11 montre un exemple d'un maillage triangulaire semi-régulier. Ce type de maillage est constitué d'un maillage grossier (polyèdre de contrôle) M_{sr}^0 et d'une hiérarchie de maillages $\{M_{sr}^l / l = 1 \dots N_l\}$ (avec N_l le nombre de résolutions), qui représente l'objet à différents niveaux de résolution. Chaque maillage M_{sr}^l est une version subdivisée du maillage représenté au niveau de résolution précédent M_{sr}^{l-1} . En effet, nous obtenons le maillage M_{sr}^l en ajoutant des nouveaux sommets au milieu de chaque arête du maillage M_{sr}^{l-1} et en les déplaçant, par la suite, afin d'approximer au mieux la surface originale. Ce processus de subdivision est illustré figure 1.11.

Ces maillages présentent les avantages suivants [111] :

- la scalabilité en résolution, qui permet la progressivité de l'affichage, de la compression, de la décompression et de la transmission ;
- l'adaptation de la résolution en fonction du support concerné ;
- la possibilité d'utiliser différents niveaux de détails, et ce, en fonction de la distance de l'objet par rapport à la caméra ;
- la possibilité de visualiser globalement l'objet, et ce, en cas de rupture lors de la compression, de la transmission ou de la décompression des détails.
- l'information de connectivité nécessaire pour passer d'une résolution à une autre est implicite. Ainsi, la seule information de connectivité indispensable est celle du maillage grossier.

En plus de la réduction au minimum de l'information de connectivité, ces maillages ont un autre avantage. Il s'agit de la "quasi-régularité" de l'échantillonnage, qui permet l'utilisation de techniques d'analyse multirésolution performantes permettant d'améliorer les taux de compression, telle que la transformée en ondelettes. En effet, l'analyse multirésolution est plus efficace sur des échantillonnages réguliers [111].

Nous avons proposé une technique originale pour le remaillage semi-régulier [70, 71]. Nous détaillerons notre contribution dans le chapitre 2 de cette première partie.

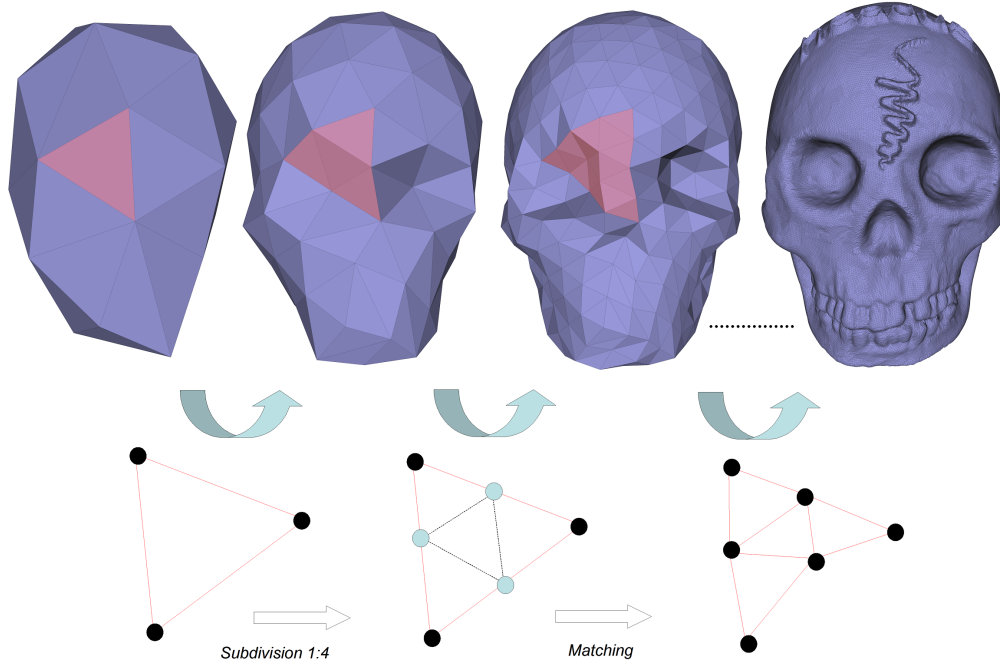


FIGURE 1.11 – Structure d'un maillage semi-régulier. En haut, de gauche à droite, nous avons le maillage de base M_{sr}^0 , le premier M_{sr}^1 , le deuxième M_{sr}^2 et le dernier $M_{sr}^{N_l}$ niveau de résolution de l'objet SKULL. En bas, la procédure de subdivision quaternaire, qui produit les différents niveaux de résolution.

1.4.2 Analyse multirésolution

L'analyse multirésolution est une alternative à la simplification séquentielle et à la décomposition spectrale pour coder les maillages de manière progressive. C'est un outil performant, qui a été utilisé depuis longtemps dans le domaine de l'imagerie et de la vidéo. La technique la plus répandue étant la transformée en ondelettes [57, 99, 107].

L'idée principale de l'analyse multirésolution pour les surfaces est de décomposer un maillage M^l en deux ensembles à l'aide de filtres d'analyse, à savoir : un maillage basse résolution M^{l-1} plus grossier, qui est obtenu à l'aide d'un filtre passe-bas, et un ensemble de détails D^{l-1} , qui est calculé dans un espace transformé à l'aide d'un filtre passe-haut (par exemple, le filtre de Butterfly [145]). En appliquant cette technique de manière itérative sur le maillage basse résolution obtenu lors de l'itération précédente, nous pouvons obtenir une représentation multirésolution du maillage original. La figure 1.12.a représente le principe de l'analyse et la figure 1.12.b illustre le principe de la synthèse.

La projection de l'information géométrique, dans un espace transformé, permet d'exploiter non seulement la corrélation spatiale, comme les méthodes utilisant une simplification séquentielle (section 1.3), mais aussi la corrélation fréquentielle. L'information géométrique est donc beaucoup plus concentrée, et cela permet déjà de réduire, sans au-

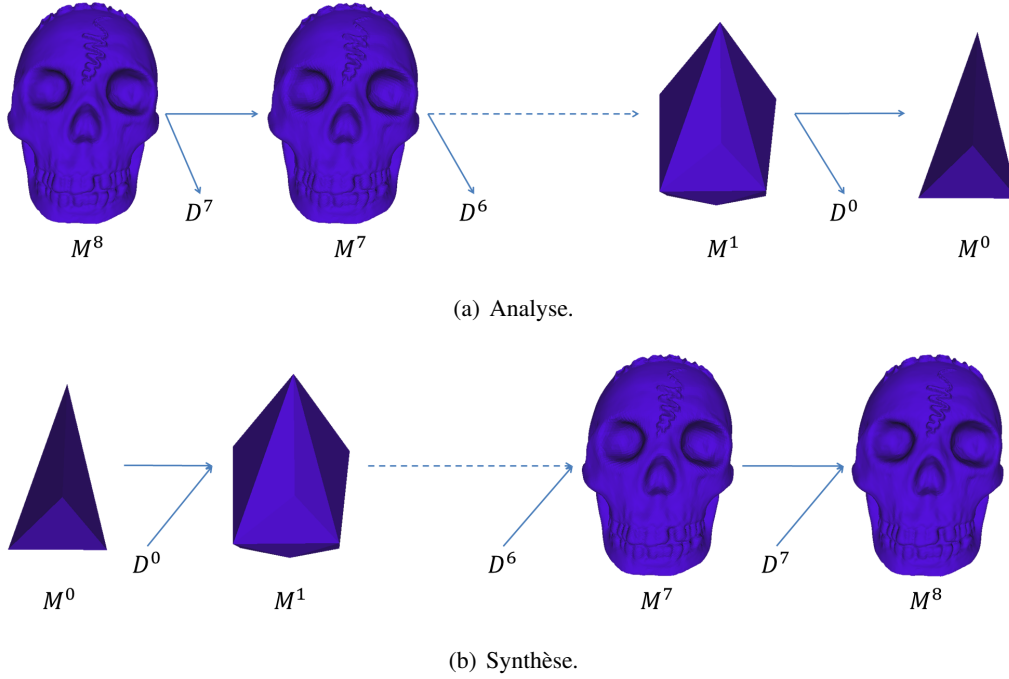


FIGURE 1.12 – Principe de l'analyse multirésolution.

cun codage particulier, l'entropie du signal [111]. De plus, l'analyse permet, par la suite, une quantification grossière des coefficients hautes fréquences (détails), en contrôlant l'impact de cette quantification sur la perception.

En utilisant le formalisme de l'analyse multirésolution, nous pouvons écrire :

$$M^l = M^{l-1} \oplus D^{l-1}, \quad (1.4)$$

avec D^{l-1} l'ensemble des détails nécessaires pour reconstruire le maillage M^l de la résolution supérieure à partir du maillage M^{l-1} et \oplus l'opérateur de complément orthogonal.

De plus, nous avons la propriété sur la notion d'espaces imbriqués propre à l'analyse multirésolution, qui nous permet d'écrire :

$$M^0 \subset M^1 \subset M^2 \subset \dots \subset M^l, \quad (1.5)$$

avec M^0 le maillage de base (résolution la plus grossière) et M^l le maillage original (résolution la plus fine).

Le principal avantage est de réduire la redondance spatio-fréquentielle, qui se présente dans les maillages. De plus, comme pour les méthodes utilisant la simplification séquentielle, ces méthodes tentent d'optimiser le compromis débit-distorsion à tout moment de la compression. Elles permettent aussi la scalabilité en résolution (pratique pour la transmission et le décodage progressif ou encore l'affichage adaptatif).

Par contre, la formulation précédente est plus facile à générer en pratique pour des signaux ayant un échantillonnage régulier, comme le son, les images et les vidéos. Dans le cas des maillages, cela sera plus difficile, car la plupart d'entre eux sont irréguliers. C'est pourquoi sont apparues des techniques de remaillage [97, 46, 56, 64, 87, 61, 88, 53, 6, 83] produisant des maillages semi-réguliers, qui ont un échantillonnage plus régulier, en vue de faciliter et d'améliorer l'efficacité de l'analyse.

Enfin, il existe la possibilité d'utiliser un outil essentiel, qui est souvent lié à l'analyse multirésolution, à savoir : l'allocation binaire. L'intérêt de cette technique est d'optimiser le compromis débit-distorsion, qui est relatif au codage des données, en répartissant l'information binaire à travers les différents niveaux de résolution en fonction de l'impact des détails sur la qualité du maillage reconstruit [30, 74, 80, 86, 91, 124, 112].

Dans le but d'améliorer les techniques existantes d'analyse multirésolution basées ondelettes, nous avons proposé un algorithme pour l'optimisation de deux transformées en ondelettes pour les maillages semi-réguliers [72, 73]. Nous détaillons notre algorithme dans le chapitre 3 de cette première partie.

1.5 Critère de qualité utilisé

Pour évaluer la qualité des maillages reconstruits après une étape de remaillage semi-régulier ou après compression, nous utiliserons la distance surface-surface. Notons que la distance surface-surface dépend de deux autres métriques, que nous appellerons la distance point-surface et la distance unilatérale entre deux surfaces [110]. Nous allons commencer par définir ces métriques.

1.5.1 Distance point-surface

La distance point-surface $d(v, M')$ entre un sommet v appartenant à un maillage M et une surface représentée par un maillage M' correspond à la distance entre v et le plus proche sommet v' de M' [110]. Son expression est donc donnée par :

$$d(v, M') = \min_{v' \in M'} \|v - v'\|_2, \quad (1.6)$$

où $\|\cdot\|_2$ dénote la norme Euclidienne L2.

1.5.2 Distance unilatérale entre deux surfaces

La distance unilatérale $\bar{d}(M, M')$ entre deux surfaces déterminées par des maillages M et M' est l'équivalent de la racine carrée de l'erreur quadratique moyenne relative à la distance point-surface [110]. Elle est donnée par l'expression suivante :

$$\bar{d}(M, M') = \sqrt{\frac{1}{|M|} \iint_{v \in M} d(v, M')^2 dM}, \quad (1.7)$$

où $|M|$ représente la surface du maillage M .

Nous appelons cette métrique distance unilatérale, car la distance $d(M, M')$ entre M et M' n'est pas forcément égale à la distance $d(M', M)$ entre M' et M , comme l'illustre l'exemple très simple de la figure 1.13 [11]. Remarquons que la distance entre le sommet A de la surface M' et la surface M est beaucoup plus grande que la distance entre B et M' . Ainsi, la distance unilatérale $d(M, M')$ sera faible, alors que la distorsion réelle entre les deux surfaces, qui es visible à l'œil nu, est beaucoup plus importante. Cette distance unilatérale n'est donc pas suffisante pour traduire la déformation d'une surface.

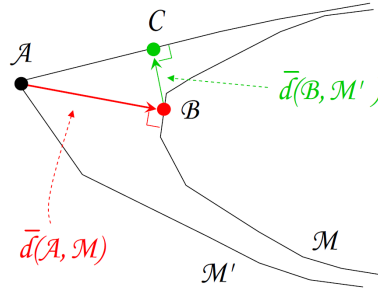


FIGURE 1.13 – Distance unilatérale entre deux surfaces. La distance $\bar{d}(M, M')$ n'est pas forcément égale à la distance $\bar{d}(M', M)$. Localement une mesure de distance point-surface peut être différente (image extraite de [111]).

1.5.3 Distance surface-surface

Pour résoudre le problème d'asymétrie de la distance unilatérale, la mesure de la distance entre deux maillages se fait généralement en prenant la valeur maximale entre $\bar{d}(M, M')$ et $\bar{d}(M', M)$, voir [110]. Dans tout ce qui va suivre, nous appellerons cette distance la distance surface-surface. Elle est définie par :

$$d_S(M, M') = \max(\bar{d}(M, M'); \bar{d}(M', M)). \quad (1.8)$$

Soulignons que cette distance est intéressante pour évaluer la distorsion entre deux maillages, puisqu'elle est indépendante de leurs topologies. D'ailleurs, nous pouvons remarquer que les trois équations (1.6), (1.7) et (1.8) ne dépendent que des sommets v appartenant aux surfaces des maillages.

1.6 Conclusion

Dans ce chapitre, nous avons présenté les maillages polygonaux et plus particulièrement les maillages semi-réguliers. Ces maillages sont très utilisés à cause de leur structure multirésolution implicite et leur échantillonnage quasi-régulier. Comme nous l'avons vu, ils permettent également l'utilisation d'une analyse multirésolution efficace, notamment les transformées en ondelettes. Cela, permet d'améliorer les taux de compression et de faciliter les transmissions progressives.

Dans le prochain chapitre, nous présentons notre première contribution, à savoir notre approche pour le remaillage semi-régulier, une étape nécessaire pour obtenir des maillages semi-régulier. Dans le chapitre 3, nous présentons notre deuxième contribution, à savoir notre algorithme d'optimisation de deux transformées en ondelettes pour les maillages semi-réguliers.

Chapitre 2

Contribution sur le remaillage : une approche basée sur la relaxation de Lloyd

2.1 Introduction

Les maillages semi-réguliers sont souvent utilisés dans le domaine émergent du traitement de la géométrie [132]. Une question fondamentale du traitement de la géométrie est la suivante : est-il possible d'étendre les méthodes classiques de traitement du signal numérique, généralement appliquées sur les structures régulières uniformes (par exemple, la transformée de Fourier discrète ou la transformée en ondelettes), à des données ayant un échantillonnage irrégulier et non uniforme ? Cette question n'a été que partiellement résolue. Par exemple, les opérateurs discrets de lissage ne sont utilisables que dans le cas d'un échantillonnage uniforme et régulier [8]. Par conséquent, la meilleure solution consiste souvent à effectuer un remaillage semi-régulier de l'objet original, qui permettra d'obtenir une approximation de ce dernier, mais avec un échantillonnage uniforme et régulier.

L'idée de base d'une méthode de remaillage semi-régulier, comme illustrée figure 2.1, est de ré-échantillonner la géométrie de la surface de l'objet tout en assurant une connectivité semi-régulière (ou connectivité de subdivision). Ceci se fait généralement en 2 étapes : une étape de simplification du maillage original afin d'obtenir un complexe de base (version grossière du maillage original), suivie d'une étape de raffinement (basée sur un schéma de subdivision quaternaire) de ce complexe de base afin d'obtenir finalement le maillage semi-régulier. Motivés par l'amélioration des techniques existantes, nous proposons dans ce chapitre notre contribution dans le domaine du remaillage semi-régulier.

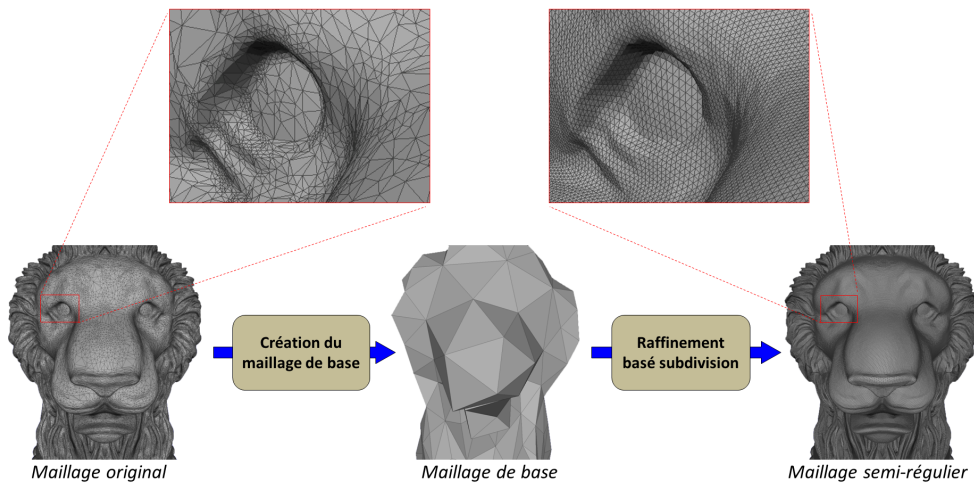


FIGURE 2.1 – Principe général du remaillage semi-régulier.

Le reste de ce chapitre est organisé de la manière suivante : la section 2.2 propose un état de l’art sur le remaillage semi-régulier. La section 2.3 présente nos motivations et nos contributions. La section 2.4 présente la méthode que nous avons développée pour le remaillage semi-régulier. La section 2.5 présente quelques résultats, et nous concluons dans la section 2.6.

2.2 Etat de l’art

Les principales techniques de remaillage semi-régulier peuvent être classées en deux catégories, selon la façon dont elles mettent en correspondance les maillages en entrée (irrégulier) et en sortie (semi-régulier). Ainsi, nous pouvons distinguer les deux catégories suivantes [8] :

1. **techniques utilisant une paramétrisation** pour mettre en correspondance les deux maillages. Ces techniques se distinguent essentiellement par le type de paramétrisation utilisé, à savoir :
 - a) **les techniques qui paramétrisent le maillage d’entrée sur un domaine global planaire.** Ce domaine est alors ré-échantillonné (une nouvelle connectivité est construite sur ce domaine) et la connectivité du nouveau maillage est projetée dans l’espace 3D, produisant ainsi une version améliorée de l’entrée ;
 - b) **les techniques qui paramétrisent le maillage d’entrée sur un ensemble de domaines triangulaires.** Cet ensemble est obtenu, soit par la simplification, soit par le partitionnement, du maillage original en un ensemble de régions (par exemple, en utilisant le partitionnement de Voronoï) ;

2. techniques utilisant une procédure autre que la paramétrisation afin de mettre en correspondance les deux maillages.

Dans les sous-sections suivantes (sections 2.2.1 et 2.2.2), nous présentons un état de l'art sur ces deux catégories de remaillage semi-régulier.

2.2.1 Remaillage semi-régulier basé paramétrisation

Une paramétrisation de la surface peut être considérée comme un "mappage" de la surface à un domaine approprié. En général, le domaine paramétrique est lui-même une surface, ce qui nous amène souvent à "mapper" une surface dans une autre. Généralement, les surfaces qui sont homéomorphes à un disque sont "mappées" dans le plan. Pour les surfaces représentées par un maillage triangulaire, les "mappages" sont linéaires par morceaux [51]. Un exemple de paramétrisation de surface est donnée figure 2.2. Tous les algorithmes présentés dans cette sous-section utilisent une paramétrisation afin de mettre en correspondance le maillage simplifié et le maillage original. Ainsi en subdivisant le maillage simplifié, on se sert de la paramétrisation pour replacer les sommets ajoutés sur la surface du maillage original.

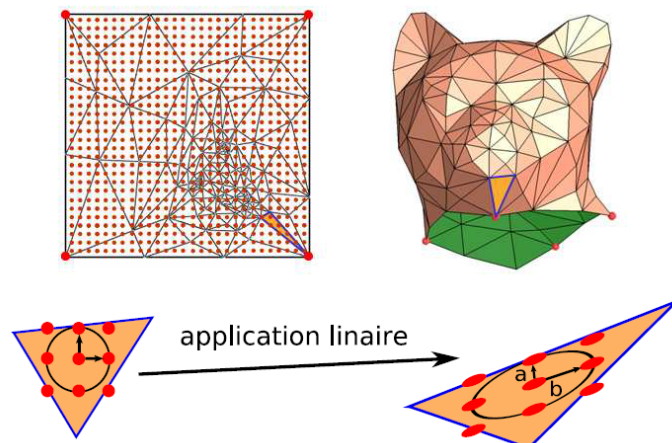


FIGURE 2.2 – Exemple de paramétrisation de surface (image tirée de [117]).

En général, une paramétrisation introduit une distorsion des angles ou des surfaces. En fait, une bonne paramétrisation tend à minimiser une certaine distorsion. Différentes façons d'y parvenir ont été proposées dans la littérature [51].

En 1993, Lounsbery *et al.* ont proposé une des premières méthodes de remaillage semi-régulier [97]. La limite principale de cette méthode est que l'étape de simplification n'est pas utilisée, en vue d'obtenir un maillage grossier M_{sr}^0 homéomorphe à la surface à remailler. Partant d'un polyèdre très simple, comme maillage grossier, cette méthode présente deux inconvénients majeurs. Tout d'abord, de nombreuses itérations sont nécessaires pour obtenir une forme qui approxime au mieux la surface originale,

sachant que pour les maillages complexes, il est souvent difficile d'y aboutir en partant d'un polyèdre simple. De plus, il faut connaître la topologie de l'objet à remailler afin d'adapter le complexe de base à chaque surface à traiter.

En 1995, Eck *et al.* ont proposé de construire le maillage grossier, en faisant une partition grossière de Voronoï de la surface initiale (basée sur les distances géodésiques) [46]. Ensuite, des cartes harmoniques sont calculées afin de "mapper" chaque région de Voronoï dans un polygone planaire. Enfin, le maillage grossier est subdivisé d'une manière itérative pour obtenir le maillage semi-régulier final.

En 1998, Lee *et al.* ont proposé un nouvel algorithme pour le remaillage semi-régulier [87]. Cet algorithme est basé sur une simplification successive des sommets du maillage irrégulier. A chaque fois qu'un sommet est supprimé, une carte conforme [43] est calculée afin de projeter le sommet supprimé sur le maillage simplifié. Ceci permet de construire une première paramétrisation, qui sera lissée grâce à un processus de subdivision utilisant le schéma de Loop [94].

En 1999, Gioia a proposé une méthode reposant sur une paramétrisation respectant les caractéristiques géométriques et les propriétés visuelles du maillage [56]. Pour cela, l'algorithme de remaillage s'appuie sur un partitionnement de la surface initiale et construit une paramétrisation préservant les zones à hautes variations, en vue d'obtenir un maillage de base plus fidèle à l'objet. Cette méthode cherche ainsi à minimiser les variations géométriques avec la surface initiale. Cela se traduit, par exemple, par une conservation des arêtes vives dans le maillage grossier de base, pour lesquelles il faut avoir théoriquement un nombre infini de coefficients pour procéder à leur raffinement, tout en partant d'une zone lisse.

En 2000, Guskov *et al.* ont proposé l'algorithme INM (*Interpolating Normal Meshes*), qui permet de créer le maillage grossier par contractions successives des demi-arêtes [64]. Durant l'étape de simplification, la paramétrisation [87] est aussi calculée. Cette paramétrisation permettra de "mapper" les arêtes du maillage grossier sur le maillage original. A l'aide de cette paramétrisation, et en utilisant la paramétrisation proposée par Floater [50], une paramétrisation lisse globale est calculée. Une fois cette paramétrisation calculée, le maillage grossier est subdivisé avec le schéma de Butterfly [145]. Avec cette méthode, chaque niveau de résolution M_{sr}^l est un déplacement normal du niveau qui le précède M_{sr}^{l-1} . Les maillages ainsi obtenus sont généralement appelés maillages "normaux" et sont particulièrement pratiques d'un point de vue compression.

En 2002, Lee *et al.* ont proposé une extension de l'algorithme INM [64] afin de pouvoir gérer les surfaces avec des bords [88].

En 2003, Khodakovsky *et al.* ont proposé un algorithme [78], qui est basé sur la même étape de simplification que celle de MAPS [87]. La paramétrisation qui en résulte est ensuite optimisée grâce à une procédure de relaxation appliquée sur l'ensemble des *patches*, mais également entre ces *patches*. Ceci permet d'obtenir une paramétrisation globalement lisse.

En 2004, Friedel *et al.* ont proposé un nouvel algorithme permettant d'obtenir des

maillages "normaux" [53]. La motivation de ces auteurs est basée sur le fait que la méthode initiale, développée par Guskov *et al.* [64], contraint un maximum de détails qui sont ajoutés entre deux résolutions à être scalaires (détails "normaux"). Ceci limite inéluctablement l'efficacité du remaillageur d'un point de vue erreur de reconstruction [117]. Il semble donc pertinent d'autoriser l'obtention de plus de détails non "normaux", si cela permet d'obtenir des maillages semi-réguliers plus fidèles aux modèles originaux d'un point de vue géométrique. Ces travaux proposent ainsi d'optimiser le compromis entre nombre maximal de détails "normaux" (hautes performances en terme de compression) et erreur minimale de remaillage, tout au long du processus (amélioration de la qualité des surfaces remaillées). L'algorithme part d'une paramétrisation globalement lisse, qui est ensuite mise à jour durant l'étape de raffinement. La mise à jour de cette paramétrisation, qui se fait par minimisation d'une fonctionnelle, permet d'obtenir, à chaque nouvelle résolution, le maillage semi-régulier qui approxime au mieux la surface originale (au sens des moindres carrés) avec un maximum de détails "normaux". Ainsi, cet algorithme permet d'obtenir des maillages semi-réguliers qui approximent les maillages originaux, et ce, à l'inverse de l'algorithme de Guskov *et al.* [64] qui utilise un schéma interpolant lors de la subdivision.

En 2006, Ahn *et al.* [6] ont proposé une implémentation *out-of-core* de MAPS [87], pour pouvoir remailler les maillages massifs, qui ne peuvent pas être chargés entièrement en mémoire. La technique utilisée consiste à charger le maillage en dehors de la mémoire principale grâce à un principe de synchronisation de *buffers* durant l'étape de simplification. Deux *buffers* sont ainsi utilisés pour stocker en mémoire, d'un côté la partie du maillage traitée, qui doit être simplifiée, et de l'autre côté les informations de paramétrisation.

Plus récemment, en 2007, Guskov a proposé un remaillageur basé sur une paramétrisation globalement lisse, qu'il a appelé TriReme [61]. Sa méthode se base sur une simplification du maillage original en utilisant le partitionnement de Voronoï et en considérant son dual pour construire le maillage grossier. Par la suite, une paramétrisation de chaque région de Voronoï sur le maillage grossier est construite. Enfin, cette paramétrisation est rendue globalement lisse, en minimisant un critère de distorsion.

2.2.2 Remaillage semi-régulier ne se basant pas sur une paramétrisation

A notre connaissance, il existe un seul travail publié dans la littérature qui propose une méthode de remaillage semi-régulier ne se basant pas sur une paramétrisation pour remplacer les sommets ajoutés par subdivision. En effet, en 1999, Kobbelt *et al.* ont proposé un algorithme de remaillage semi-régulier qui simule le processus de *Shrink Wrapping*, en utilisant la technique de surfaces déformables [83]. L'algorithme résultant génère des maillages semi-réguliers ayant un maillage grossier défini par un nombre réduit de triangles. Il utilise également, un processus d'optimisation itératif qui distribue les sommets du maillage semi-régulier sur la surface originale, garantissant ainsi, une

faible distorsion locale.

2.3 Motivation et contribution

Nous proposons ici un algorithme original pour le remaillage semi-régulier basé sur le diagramme de Voronoï. L'originalité de notre méthode réside dans le fait que nous utilisons le diagramme de Voronoï, non seulement durant l'étape de simplification, mais aussi durant l'étape de raffinement (voir figure 2.3) :

- **étape de simplification**, durant cette étape nous calculons un diagramme de Voronoï centroïdal sur les points du maillage irrégulier M_{ir} . Les sites des régions de ce diagramme seront les sommets du maillage de base. Nous testerons par la suite deux méthodes différentes pour cette étape ;
- **étape de raffinement**, durant cette étape nous subdivisons, de manière itérative, le maillage de base, qui est obtenu dans l'étape de simplification. Les sommets ajoutés par subdivision sont considérés comme des nouveaux sites de Voronoï, et une relaxation de Lloyd est appliquée pour mettre à jour le diagramme de Voronoï.

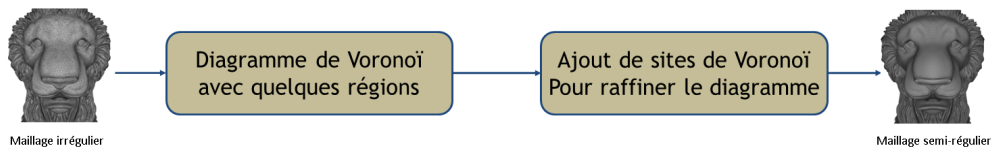


FIGURE 2.3 – Schéma général de notre méthode.

Nous proposons aussi un critère en nous basant sur la masse des sommets du maillage d'entrée, qui permet de raffiner seulement les triangles se trouvant dans les régions détaillées. Ceci permettra d'obtenir des maillages semi-réguliers adaptatifs.

2.4 Méthode proposée

Dans cette section, nous commençons par introduire dans la section 2.4.1 les diagrammes de Voronoï. Ensuite, nous présentons les diagrammes de Voronoï centroïdaux dans la section 2.4.2. Nous présentons par la suite les deux étapes majeures de notre méthode dans la section 2.4.3. Ensuite, nous donnons quelques résultats expérimentaux dans la section 2.5. Enfin nous concluons dans la section 2.6.

2.4.1 Diagramme de Voronoï

L'origine du diagramme de Voronoï remonte au 17ème siècle. Dans son livre sur les principes de la philosophie, R. Descartes [36] affirme que le système solaire est constitué de tourbillons. Ses illustrations montrent une décomposition de l'espace en régions convexes, qui sont composées chacune d'elles de matière tournante autour des étoiles fixes (voir figure 2.4).

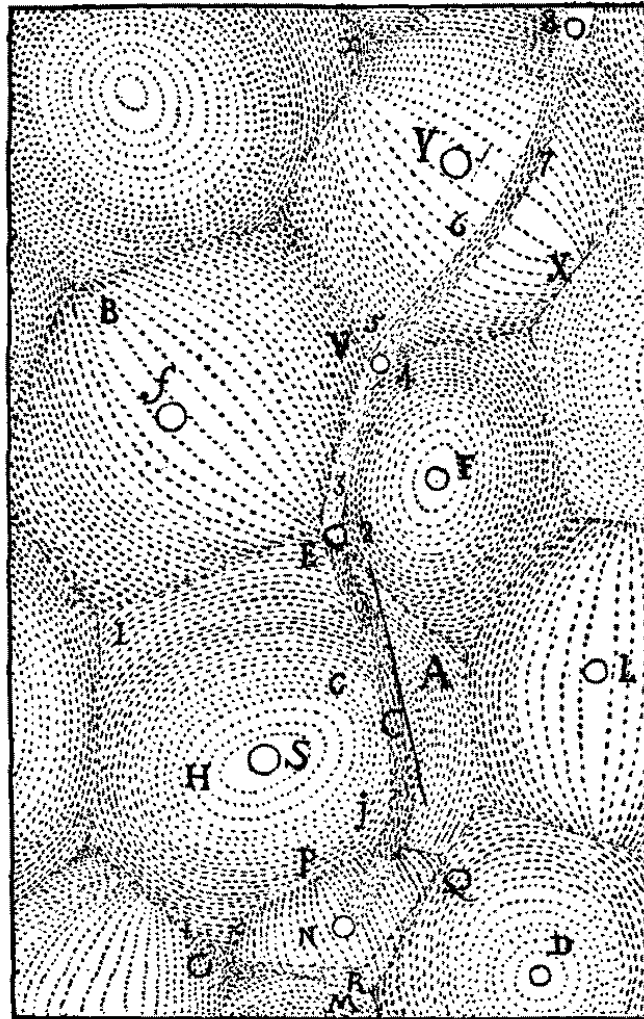


FIGURE 2.4 – Décomposition de l'espace en tourbillons par Descartes.

Même si Descartes n'a pas explicitement défini l'extension de ces régions au diagramme de Voronoï, l'idée sous-jacente semble être la suivante :

- soient un espace $M = \{v\}$ et un ensemble de sites $S = \{p\}$, tel que : $S \subset M$, avec une notion d'influence exercée par les sites p sur les points v de M ;
- la région d'un site p sera alors composée par les points v , pour lesquels l'influence de p est la plus forte.

Ce concept, qui a été émergé d'une façon indépendante, s'est avéré utile dans divers domaines scientifiques. Différents noms ont été utilisés pour ce concept, tels que l'axe médian (en biologie et en physiologie), les zones de Wigner-Seitz (en chimie et en physique), les domaines d'action en cristallographie et les polygones de Thiessen (en météorologie et en géographie). Les mathématiciens Dirichlet [38] et Voronoï [139] ont

été les premiers à présenter formellement ce concept. Ils l'ont utilisé pour l'étude des formes quadratiques. Dans leurs formalisations, l'ensemble des sites est un réseau de points entier, et l'influence est mesurée par la distance Euclidienne. La structure résultante a été appelée partitionnement de Dirichlet ou diagramme de Voronoï.

En 1908, Voronoï [139] a été le premier à considérer le dual de cette structure, où chaque deux sites sont connectés si leurs régions ont une frontière commune. Plus tard, Delaunay [34] a obtenu le même résultat, où il annonce que deux sites sont connectés si et seulement si ils se trouvent sur un cercle, dont l'intérieur ne contient aucun autre site. Depuis, le dual du diagramme de Voronoï est appelé la triangulation de Delaunay.

Outre ses applications dans d'autres domaines scientifiques, le diagramme de Voronoï et son dual peuvent être utilisés pour résoudre différents problèmes géométriques. Par ailleurs, ces structures sont très attrayantes, et beaucoup de recherches ont été consacrées à leurs études, et ce, depuis que Shamos *et al.* [123] l'ont introduit dans le domaine de la géométrie algorithmique (environ un travail publié sur seize de la géométrie algorithmique fait l'étude des diagrammes de Voronoï [12]).

Définition et notations utilisées

Soient :

- $\mathcal{S} \subset \mathbb{R}^n$ un ensemble ouvert,
- $\{s_k; k = \{0, 1, \dots, K-1\}\}$ K sites différents.

Le diagramme de Voronoï peut être défini par l'ensemble de K régions distinctes R_k , tel que :

$$R_k = \{v \in \mathcal{S} \mid d(v, s_k) \leq d(v, s_j); j \neq k\}, \quad (2.1)$$

avec d une mesure de distance donnée.

Un exemple de diagramme de Voronoï en 2D est donné figure 2.5.

2.4.2 Diagramme de Voronoï centroïdal

Un diagramme de Voronoï centroïdal est un diagramme de Voronoï, tel que chaque site s_k est le centre de masse de la région R_k [42], soit :

$$s_k = \frac{\int_{R_k} v \rho(v) dv}{\int_{R_k} \rho(v) dv}, \quad (2.2)$$

où $\rho(v)$ est une fonction de masse.

Pour un ensemble de sites $\{v_k\}$ donnés, un diagramme de Voronoï centroïdal est le diagramme qui minimise l'énergie $E(\{(R_k, s_k)\})$ suivante :

$$E = \sum_{k=0}^{K-1} \int_{R_k} \rho(v) \|v - s_k\|^2 dv. \quad (2.3)$$

Ces diagrammes sont souvent utilisés dans les applications qui nécessitent un bon échantillonnage des données traitées. Plusieurs méthodes (déterministes et probabilistes)

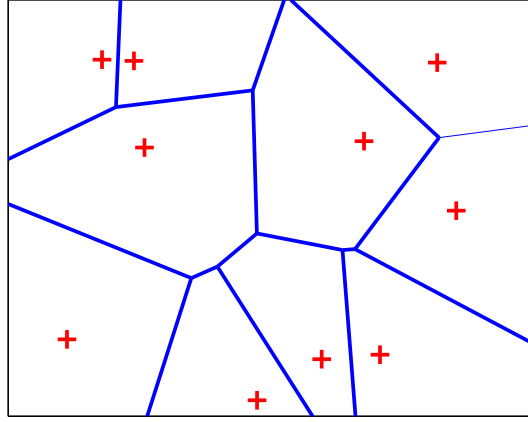


FIGURE 2.5 – Exemple de diagramme de Voronoï en 2D : les + en rouge représentent les sites s_k .

existent pour la construction des diagrammes de Voronoï centroïdaux [42]. La relaxation de Lloyd [93] est la méthode déterministe la plus connue.

Relaxation de Lloyd

La relaxation de Lloyd est un algorithme de classification de données, qui permet aussi la construction d'un diagramme de Voronoï centroïdal. On définit :

- un ensemble ouvert $\mathcal{S} \subset \mathbb{R}^n$;
- un ensemble initial de K sites différents $\{s_k \in \mathcal{S}; k = \{0, 1, \dots, K-1\}\}$;
- un diagramme de Voronoï initial caractérisé par K régions distinctes R_k , tel que : $R_k = \{v \in \mathcal{S} \mid d(v, s_k) \leq d(v, s_j); j \neq k\}$, où d est une mesure de distance donnée ;
- une distorsion initiale $D_T = \sum_{k=1}^K \int_{R_k} \rho(v) \|v - s_k\|^2 dv$, qui est relative à ce diagramme.

L'algorithme itératif de Lloyd est constitué par les trois étapes suivantes :

- recalculer les sites $s_k = \frac{1}{|R_k|} \int_{v \in R_k} \rho(v) v$, où $|R_k|$ représente le nombre de points v appartenant à la région R_k . Ainsi, chaque site s_k sera égal au centre de masse de sa région R_k ;
- recalculer le diagramme de Voronoï, qui est relatif aux nouveaux sites ;
- recalculer la distorsion D_T .

L'algorithme s'arrête quand la variation de distorsion entre deux itérations successives est inférieure à un certain seuil, qui est donné par l'utilisateur. Un aperçu de la relaxation de Lloyd est donné par l'algorithme 1 et un exemple est donné figure 2.6.

Algorithm 1 Relaxation de Lloyd

```

{seuil d'arrêt qui est défini par l'utilisateur, ici il est égal à 0,01 :}
seuil  $\leftarrow$  0,01
{initialisation :}
sites =  $\{s_0, s_1, \dots, s_{K-1}\}$ 
distorsioninitiale  $\leftarrow$  0
for all  $v \in \mathcal{S}$  do
     $s \leftarrow \text{PlusProcheSite}(v, \text{sites})$ 
    region( $s$ ).push_back( $v$ )
    distorsioninitiale  $\leftarrow$  distorsioninitiale +  $\rho(v)d(v, s)$ 
end for
distorsion  $\leftarrow$  distorsioninitiale
{Relaxation :}
repeat
    distorsioninitiale  $\leftarrow$  distorsion
    distorsion  $\leftarrow$  0
    for all site  $\in$  sites do
        site  $\leftarrow$  CentreDeMasse(region(site))
        region(site).clear()
    end for
    for all  $v \in \mathcal{S}$  do
         $s \leftarrow \text{PlusProcheSite}(v, \text{sites})$ 
        region( $s$ ).push_back( $v$ )
        distorsion  $\leftarrow$  distorsion +  $\rho(v)d(v, s)$ 
    end for
until (distorsioninitiale - distorsion)/distorsioninitiale  $\leq$  seuil

```

Diagramme de Voronoï centroïdal pour les maillages surfaciques

Dans notre cas, le diagramme de Voronoï est construit sur des maillages surfaciques. Par conséquent,

- l'ensemble \mathcal{S} est l'ensemble de sommets v_i du maillage irrégulier M_{ir} ;
- la masse ρ d'un sommet v_i est la surface de sa cellule duale, comme illustre la figure 2.7 ;
- la distance d est la distance Euclidienne ;
- les sites s_k sont remplacés sur la surface de M_{ir} (en cherchant le sommet le plus proche appartenant à M_{ir}), et ce, pour garantir que les sites resteront toujours sur la surface du maillage original.

La figure 2.8 montre un exemple de diagramme de Voronoï, qui est construit sur VENUS.

Nous avons choisi de définir la masse ρ d'un sommet v_i par la surface de sa cellule duale, afin de donner plus d'importance aux sommets ayant des grands triangles adjacents. En effet, un maillage qui est représenté par un ensemble fini de sommets est un échantillonnage d'une surface continue. Par suite, il faut prendre en compte les zones du maillages ne contenant pas de sommets en attribuant à chaque sommet une masse

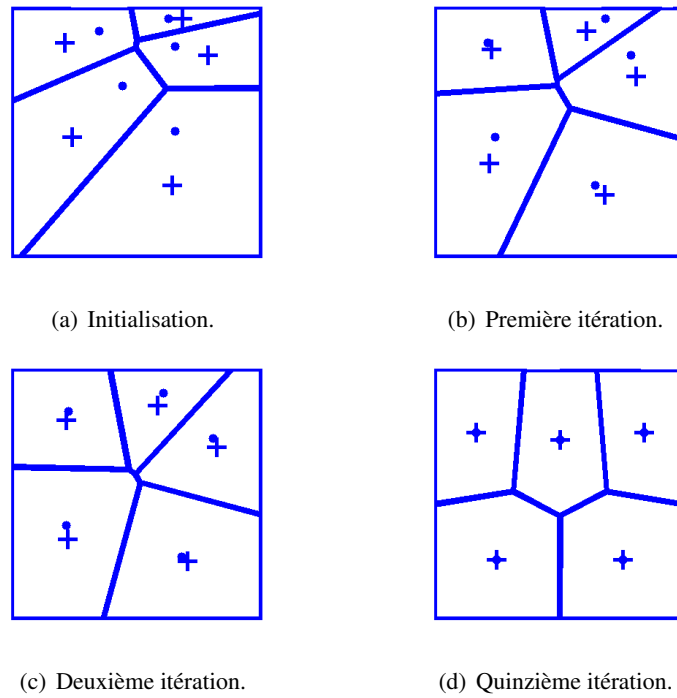


FIGURE 2.6 – Exemple de la relaxation de Lloyd : les + représentent les centres de masse des différentes régions et les points représentent les sites de Voronoï (images extraites de [1]).

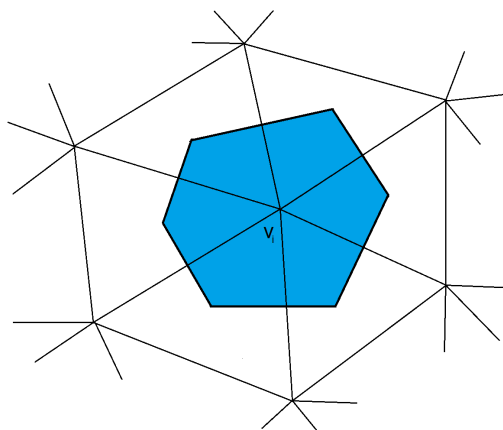


FIGURE 2.7 – Voisinage d'un sommet v_i . La cellule en bleu est le dual de v_i . La surface de cette cellule représente la masse $\rho(v_i)$.



FIGURE 2.8 – Exemple de diagramme de Voronoï, qui est construit sur VENUS.

relative à sa cellule duale. Ceci, nous permet en l'occurrence d'avoir des régions de Voronoï ayant des surfaces similaires. La figure 2.9 montre l'impact de cette masse sur la construction d'un diagramme de Voronoï.

2.4.3 Description de l'algorithme proposé

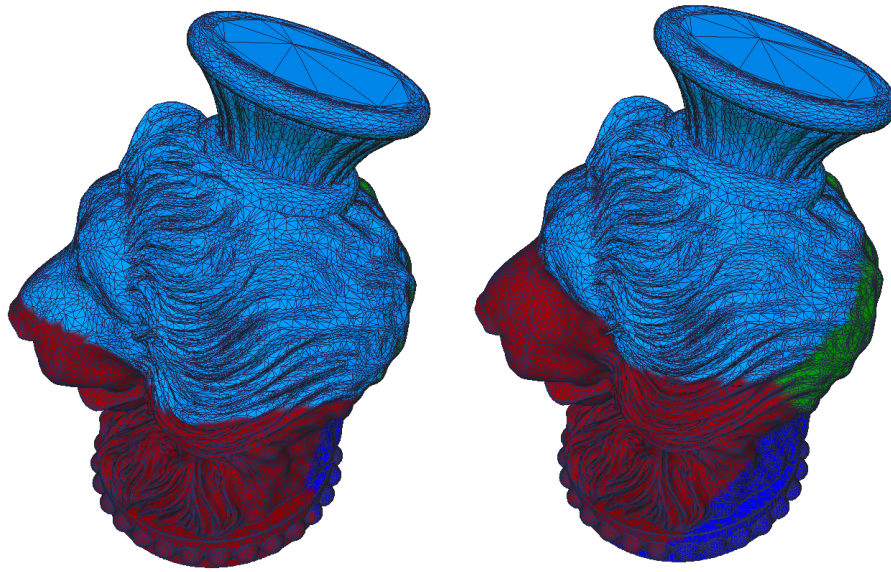
Dans cette sous-section, nous décrivons les deux étapes majeures de notre méthode, à savoir : la simplification et le raffinement.

Par la suite, nous noterons par s_k^l un site s_k , qui est donné à une résolution l .

Etape de simplification

Le but de l'étape de simplification est de produire le maillage de base M_{sr}^0 à partir du maillage irrégulier M_{ir} . Notons par s_k^l le $k^{\text{ème}}$ site de Voronoï à une résolution l . L'objectif visé ici est de créer un diagramme de Voronoï avec quelques sites s_k^0 . Ces sites correspondent aux sommets de M_{sr}^0 . Pour déterminer ces sites, nous avons testé les deux méthodes suivantes :

1. la méthode itérative de LBG (Linde–Buzo–Gray), qui est proposée par Linde *et al.* [92]. L'algorithme LBG consiste à créer un diagramme de Voronoï initial, qui est composé par une seule région, puis itérer le processus suivant jusqu'à l'obtention du nombre de sites souhaité, soient :



(a) Diagramme de Voronoï obtenu sans prendre en compte la masse ρ .

(b) Diagramme de Voronoï obtenu en prenant en compte la masse ρ .

FIGURE 2.9 – Impact de la masse ρ sur la construction d'un diagramme de Voronoï. Nous remarquons qu'en prenant en compte la masse ρ lors de la construction du diagramme de Voronoï, les surfaces des différentes régions tendent à être similaires. Nous pouvons remarquer que vu qu'il n'y pas assez de sommets dans la base de l'objet VASE LION, la région qui contient la base a une surface plus grande que les autres quand on prend pas en compte la masse ρ .

- diviser le site s_k^0 de chaque région R_k^0 en deux sites (le nombre de régions étant ainsi multiplié par deux),
- utiliser la relaxation de Lloyd jusqu'à convergence pour mettre à jour le diagramme.

La figure 2.10 montre différentes itérations de l'algorithme LBG [92] utilisé sur l'objet VENUS.

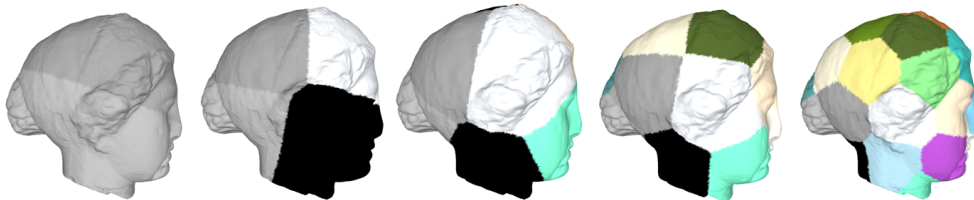


FIGURE 2.10 – Différentes itérations de l'algorithme LBG [92] appliquées sur l'objet VENUS.

2. une méthode basée sur la décimation de sommets en fonction de leurs courbures présenté dans [87]. Pour cela, nous commençons par attribuer à chaque sommet v_i

du maillage original un poids $w(v_i)$ en fonction de sa courbure moyenne et de sa masse $\rho(v_i)$. Ce poids $w(v_i)$ est donné par l'équation suivante :

$$w(v_i) = 0.5 \frac{\rho(v_i)}{\max_{v_k \in M_{ir}} \rho(v_k)} + 0.5 \frac{\kappa(v_i)}{\max_{v_k \in M_{ir}} \kappa(v_k)}, \quad (2.4)$$

avec $\kappa(v_i)$, la courbure de v_i . Pour estimer cette courbure, nous avons utilisé la méthode de Meyer *et al.* proposée dans [106].

Une fois les poids calculés pour chaque sommet, nous procédons à la décimation des sommets, en commençant par celui qui a le poids le plus faible, jusqu'à obtenir le nombre souhaité de sommets pour le maillage grossier. L'algorithme 2 montre cette procédure de décimation de sommets.

Algorithm 2 Décimation de sommets

```

M ← Mir
while NombreTriangles(M) < nombresouhait do
  for all v ∈ M do
    {heap est une arbre binaire contenant les sommets de M et ordonnée selon les
    poids de ces sommets}
    heap.clear()
    heap.Add(v, CalculerPoids(v))
  end for
  while heap n'est pas vide do
    M.SupprimerSommet(heap.Pop())
    Enlever les sommets voisins du sommet supprimé de l'arbre binaire heap
    Trianguler le trou généré dans le maillage M par cette suppression
  end while
end while

```

Une fois la décimation finie, nous utilisons la relaxation de Lloyd sur le maillage irrégulier M_{ir} , avec comme initialisation l'ensemble des sommets obtenus après la procédure de décimation, afin d'obtenir le diagramme grossier et les sites de ce diagramme.

Construction du maillage grossier à partir des sites sélectionnés

Une fois les sites sélectionnés, les sommets de M_{ir} restants (qui ne sont pas des sites de Voronoï) sont supprimés. Pour ceci, nous supprimons un à un tout les sommets de M_{ir} qui ne sont pas des sites de Voronoï, et à chaque fois qu'un sommet est supprimé, nous fermons le trou résultant avec une triangulation de Delaunay. Ainsi nous obtenons le maillage de base M_{sr}^0 comme le montre la figure 2.11.

Durant ce processus, la paramétrisation MAPS proposée dans [87] est aussi calculée. Cette paramétrisation sert à projeter les sommets supprimés du maillage original sur le maillage grossier M_{sr}^0 et elle sera utilisée durant l'étape de raffinement.

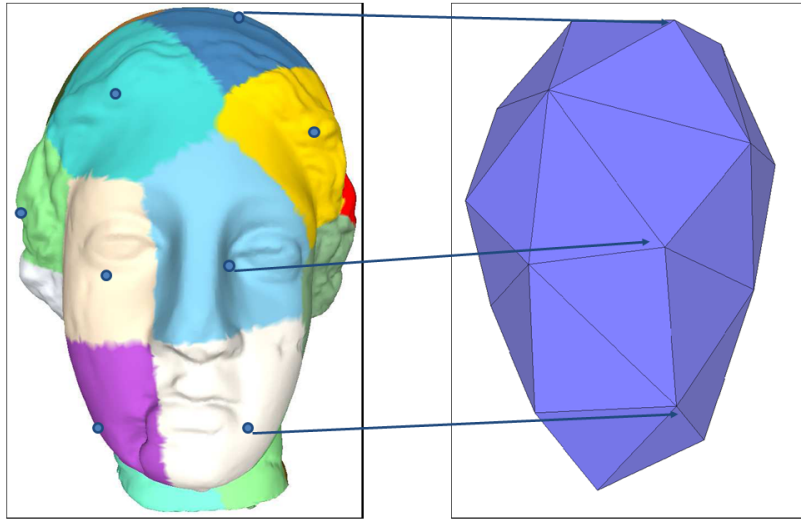
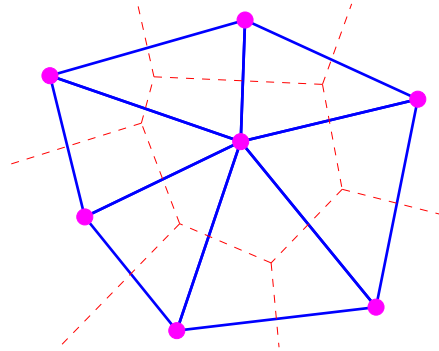


FIGURE 2.11 – A gauche : diagramme de Voronoï construit sur VENUS. A droite : version simplifiée utilisée comme maillage de base pour le maillage semi-régulier M_{sr} .

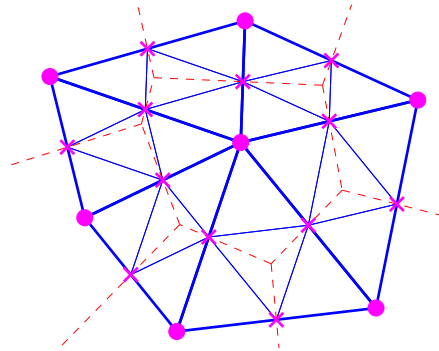
Raffinement uniforme

L'objectif visé dans l'étape de raffinement est de produire un maillage semi-régulier M_{sr} à partir du maillage grossier M_{sr}^0 . Notons toutefois que cette étape de raffinement se fait de manière itérative. Pour une itération donnée, la procédure permettant d'obtenir le maillage M_{sr}^l à partir de M_{sr}^{l-1} est la suivante (voir figure 2.12) :

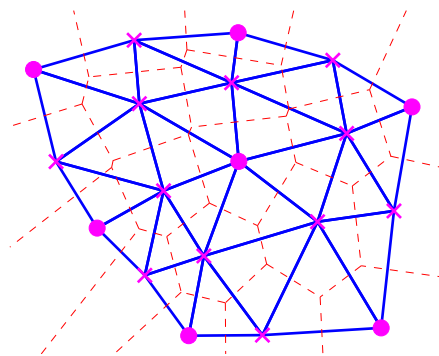
- le maillage M_{sr}^{l-1} est subdivisé (en utilisant la subdivision quaternaire), en associant un sommet à chacune de ses arêtes (voir figure 2.12(b)) ;
- chaque nouveau sommet est considéré comme un nouveau site de Voronoï s_k^l . Sa position est initialisée par la projection du milieu de l'arête, à laquelle il est associé, sur la surface originale. Cette initialisation est possible grâce à la paramétrisation que nous avons calculé durant l'étape de simplification. En effet, nous avons tout les sommets du maillage original projetés sur le maillage grossier. Pour replacer un site sur la surface originale, nous cherchons parmi tout les sommets projetés celui qui est le plus proche le sommet le plus proche au milieu de l'arête. Afin d'accélérer la recherche du sommet le plus proche, nous utilisons des *kdtree* ;
- la relaxation de Lloyd est finalement appliquée sur l'ensemble des sites, en vue d'obtenir le maillage semi-régulier M_{sr}^l (voir figure 2.12(c)). Notons qu'à la fin de la relaxation de Lloyd, nous remplaçons tout les sites sur la surface originale, en cherchant pour chaque site, le sommet le plus proche parmi les sommets du maillage original. Nous utilisons aussi des *kdtree* afin d'accélérer ce remplacement.



(a) En bleu : un ensemble de triangles à une résolution $(l - 1)$. En rouge : le diagramme de Voronoï.



(b) Les triangles sont subdivisés ; chaque nouveau sommet (représenté par '+') est un nouveau site de Voronoï.



(c) Le diagramme de Voronoï est mis à jour avec la relaxation de Lloyd.

FIGURE 2.12 – Principe d'une itération de notre étape de raffinement.

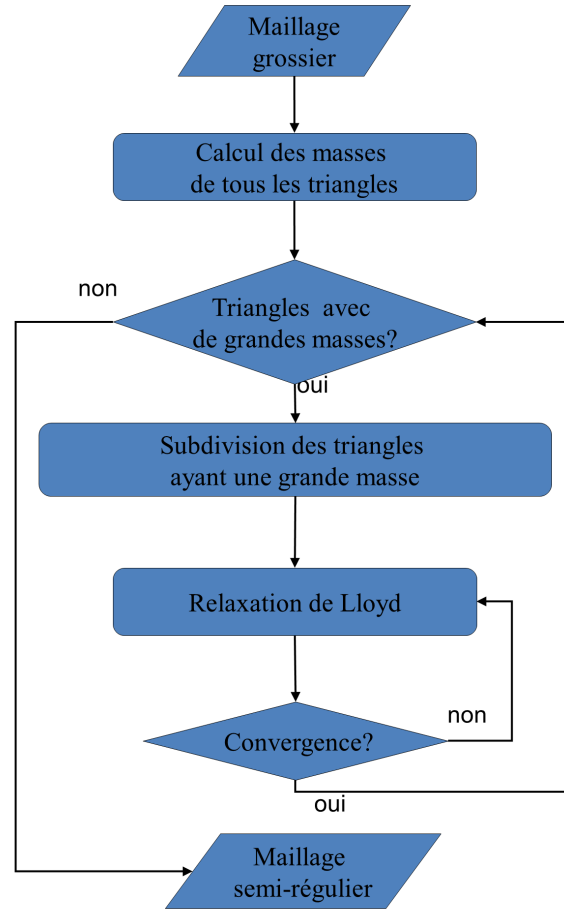


FIGURE 2.13 – Etape de raffinement proposée.

Raffinement adaptatif

Nous pouvons indiquer la limitation d'utiliser un raffinement uniforme : la subdivision uniforme de tous les triangles n'est pas toujours un élément pertinent. En effet, il n'est pas forcément utile de subdiviser les triangles qui sont dans des régions plates, contrairement aux triangles qui sont dans des régions très détaillées. La résolution de chaque triangle devrait donc être adaptée aux caractéristiques géométriques locales de la surface d'entrée. Par conséquent, nous proposons de rendre cet algorithme adaptatif. Un résumé de la technique proposée pour faire du raffinement adaptatif est donné figure 2.13. Pour cela, nous définissons, à chaque niveau de résolution l , une fonction de masse f , qui est associée à chaque site s_k^l , telle que :

$$f(s_k^l) = |R_k^l|, \quad (2.5)$$

où $|R_k^l|$ est le nombre de sommets de M_{ir} associés à R_k^l (la $k^{\text{ième}}$ région de Voronoï à la résolution l).

Ensuite, nous associons à chaque triangle t^l de M_{sr}^l , qui est composé par trois sites de Voronoï $s_{k_1}^l$, $s_{k_2}^l$ et $s_{k_3}^l$, une fonction de masse g , telle que :

$$g(t^l) = \inf_{q \in \{k_1, k_2, k_3\}} (f(s_q^l)). \quad (2.6)$$

Une fois les masses calculées, uniquement les triangles avec une masse supérieure à un seuil de subdivision ϵ donné par l'utilisateur seront subdivisés. Cette procédure devrait nous permettre de subdiviser uniquement les régions détaillées, qui contiennent généralement plus de sommets que dans les régions plates. En effet, généralement, lors de la conception d'un objet 3D, on met plus de sommets dans les régions détaillées que dans les régions plates, comme le montre la figure 2.14.

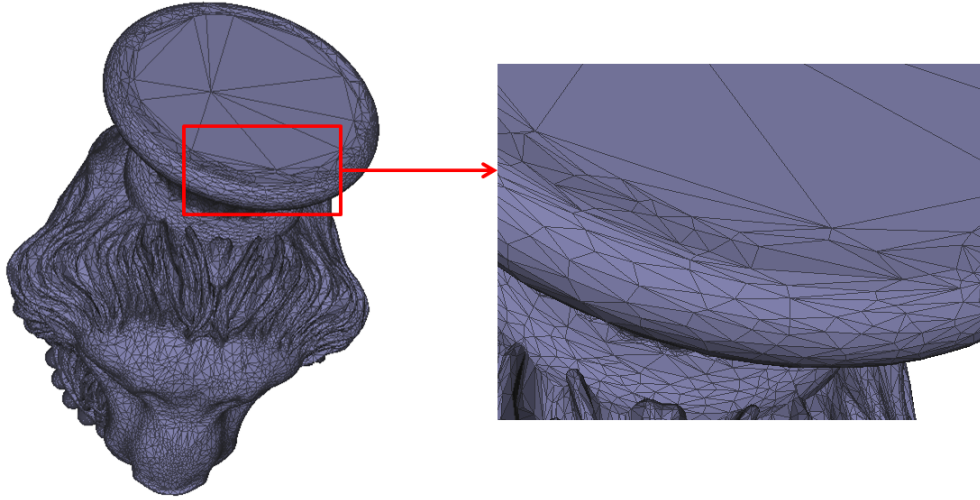


FIGURE 2.14 – Echantillonnage de l'objet VASELION. Nous remarquons que sur la base (région plate), il y a beaucoup moins de triangles que sur les régions détaillées.

2.5 Résultats expérimentaux

2.5.1 Objets utilisés

Le tableau 2.1 donne les caractéristiques des différents objets utilisés lors de nos expérimentations.

2.5.2 Pré-traitement des maillages d'entrée

Lors de la construction d'un diagramme de Voronoï, l'utilisation d'une distance géodésique semble être plus pertinente que l'utilisation de la distance Euclidienne (comme nous l'avons proposé dans la section 2.4.2). Mais, d'un autre côté, le biais introduit par

TABLE 2.1 – Caractéristiques des différents objets utilisés.

Objet	Nombre de sommets	Nombre de triangles
BIMBA	74764	149524
HORSE	48485	96966
RABBIT	67039	134074
SKULL	20002	40000
VENUS	50002	100000

la distance Euclidienne est faible quand nous utilisons des maillages fortement échantillonnés [135]. Dans notre cas, la construction du diagramme de Voronoï est efficace si le maillage M_{ir} est suffisamment dense par rapport au maillage M_{sr} . En effet, le maillage semi-régulier M_{sr} est obtenu à partir de la construction d'un diagramme de Voronoï sur le maillage irrégulier M_{ir} . Ce diagramme ne peut pas avoir plus de sommets que le nombre de sommets du maillage irrégulier M_{ir} . Par conséquent, le nombre de sommets du maillage semi-régulier obtenu est limité par l'ensemble des sommets du maillage irrégulier, ce qui n'est pas suffisant pour avoir une bonne approximation de l'objet irrégulier original. Donc, si le maillage irrégulier en entrée n'est pas suffisamment dense, il faut passer par une étape de sur-échantillonnage du maillage irrégulier.

Pour cette étape de sur-échantillonnage, nous avons testé les deux schémas suivants de subdivision :

- le schéma approximant de Loop [94],
- le schéma interpolant de Butterfly [45].

Normalement, nous devons obtenir de meilleurs résultats avec le schéma de Butterfly [45] parce qu'il est interpolant et par suite il nous permet de se rapprocher au mieux de la surface limite qui est définie par le maillage originale. Les résultats expérimentaux ont confirmé nos attentes et la figure 2.15 montre les courbes d'évolution de l'erreur de remaillage (voir chapitre 1, section 1.5) durant l'étape de raffinement pour l'objet VENUS lorsque nous sur-échantillonnons l'objet irrégulier en se basant sur les schémas de Loop ou de Butterfly. Chaque point de la courbe correspond à une itération durant l'étape de raffinement. Nous remarquons que l'erreur de remaillage est plus faible quand l'objet original est sur-échantillonné avec le schéma de subdivision de Butterfly. Ainsi, pour les résultats qui suivent, les objets irréguliers sont sur-échantillonnés avec le schéma de subdivision de Butterfly avant d'appliquer notre technique de remaillage.

2.5.3 Impact du seuil de subdivision ϵ

Afin d'étudier l'impact du seuil de subdivision ϵ utilisé au cours de l'étape de raffinement, nous avons remaillé VENUS avec différentes valeurs de ϵ . La figure 2.16 montre les courbes l'évolution de l'erreur de remaillage (voir chapitre 1, section 1.5). Nous remarquons qu'en augmentant le seuil de subdivision, l'erreur de remaillage décroît plus rapidement. Ceci est dû au fait qu'on subdivise uniquement les régions les plus dé-

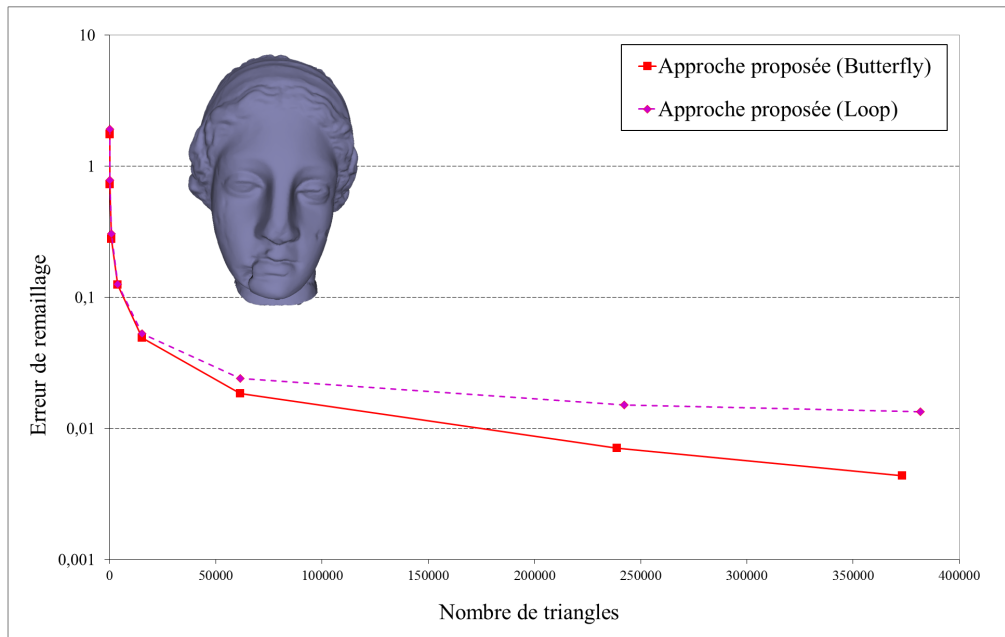


FIGURE 2.15 – Courbes d’évolution de l’erreur de remailage (échelle logarithmique) pour l’objet VENUS durant l’étape de raffinement. Le maillage irrégulier a été sur-échantillonné avec le schéma de Loop (ligne pointillée) et le schéma de Butterfly (ligne continue).

taillées. Ces résultats obtenus confirment bien l’intérêt de notre seuillage lors de l’étape de raffinement.

2.5.4 Impact de la méthode de sélection des sites du maillage grossier

Pour déterminer les sites du maillage grossier, nous avons testé les deux méthodes suivantes :

- la méthode itérative de LBG (Linde–Buzo–Gray), qui est proposée par Linde *et al.* [92], qui est une méthode qui ne prend pas en compte les propriétés géométriques des sommets,
- une méthode basée sur la décimation de sommets en fonction de leurs courbures présenté dans [87].

Les figures 2.17 et 2.18 montrent les courbes l’évolution de l’erreur de remailage pour les objets SKULL et BIMBA. Nous remarquons que nous obtenons une plus faible erreur en utilisant la méthode basée sur la décimation de sommets en fonction de leur courbure. Ceci s’explique par le fait que l’initialisation des sites est très importante lors de la construction d’un diagramme de Voronoï centroïdal, et le fait de les initialiser en tenant compte des propriétés géométriques des maillages, nous permet de mieux préserver l’allure des maillages à basse résolution. Ainsi, pour les résultats qui suivent, nous utilisons la méthode basée sur la décimation de sommets en fonction de leur courbure pour la sélection des sites du maillage grossier.

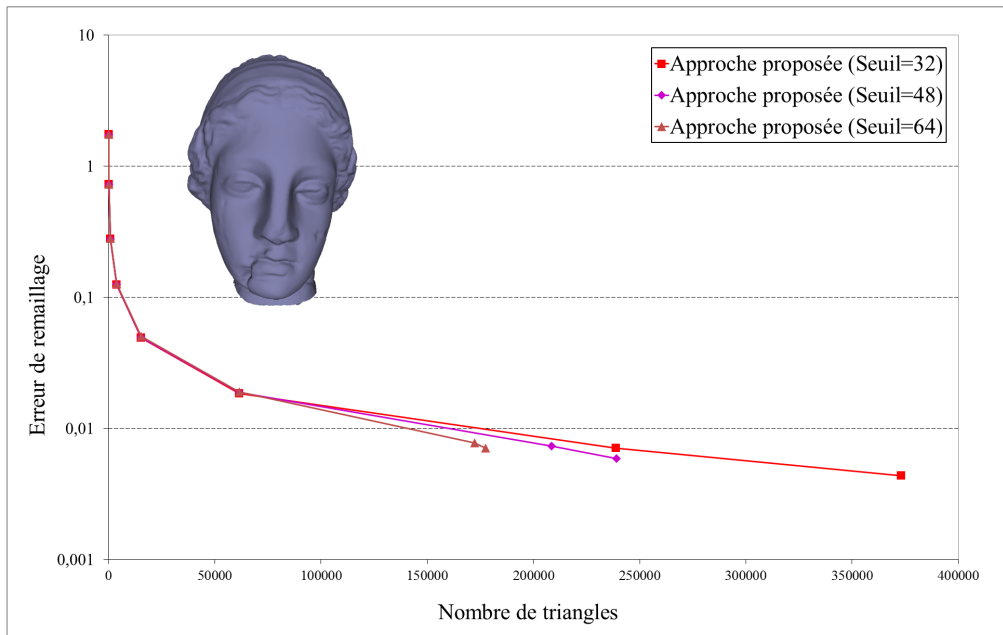


FIGURE 2.16 – Courbes d'évolution de l'erreur de remaillage (échelle logarithmique) pour l'objet VENUS durant l'étape de raffinement en fonction du seuil de subdivision utilisé durant l'étape de raffinement.

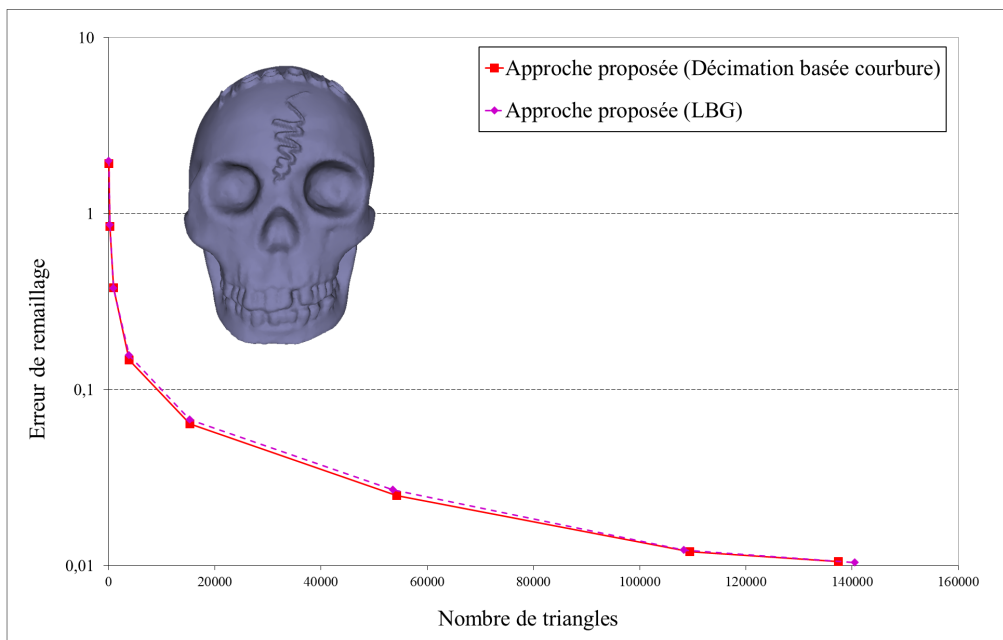


FIGURE 2.17 – Courbes d'évolution de l'erreur de remaillage (échelle logarithmique) pour l'objet SKULL durant l'étape de raffinement en fonction de la méthode de simplification utilisée.

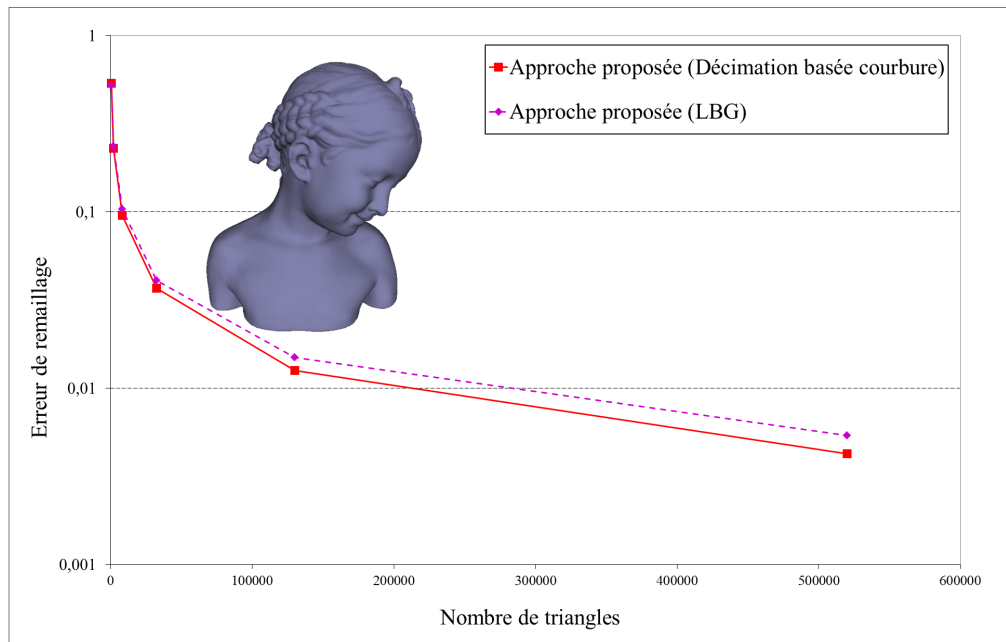


FIGURE 2.18 – Courbes d'évolution de l'erreur de remaillage (échelle logarithmique) pour l'objet BIMBA durant l'étape de raffinement en fonction de la méthode de simplification utilisée.

2.5.5 Résultats visuels

La figure 2.19 montre des résultats visuels pour l'objet SKULL. Nous remarquons que l'objet remaillé avec notre approche adaptative donne une meilleure qualité visuelle, tout en réduisant le nombre de triangles dans les régions lisses. Ajoutons aussi, qu'avec notre méthode, les régions détaillées (par exemple, la cicatrice) contiennent plus de triangles par rapport aux régions lisses.

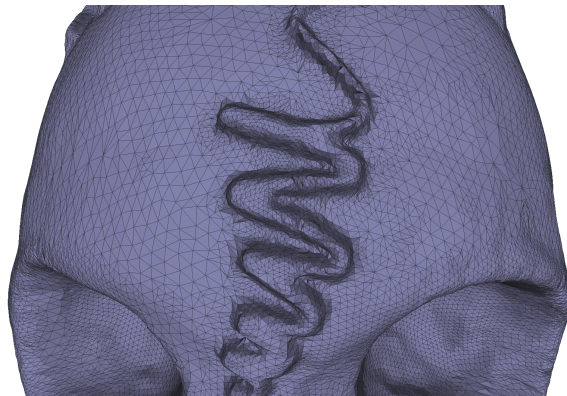
Les figures 2.20, 2.21, 2.22 et 2.23 donnent des résultats visuels additionnels pour les objets BIMBA, HORSE, RABBIT et VENUS respectivement.

2.5.6 Comparaison avec l'état de l'art

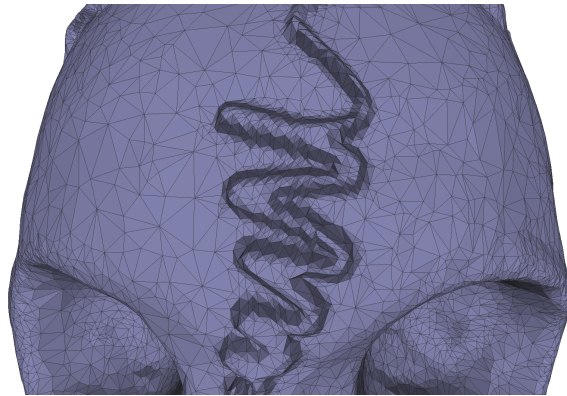
Nous avons aussi comparé nos résultats avec des résultats obtenus avec les remaillieurs développés dans [87, 64, 61]. Les figures 2.24, 2.25, 2.26 et 2.27 montrent les courbes d'évolution de l'erreur de remaillage pour VENUS, SKULL, RABBIT, BIMBA et HORSE. Nous remarquons que, pour la même erreur de remaillage, notre approche réduit fortement le nombre de triangles, qui sont nécessaires pour la construction du maillage semi-régulier.

2.5.7 Discussion

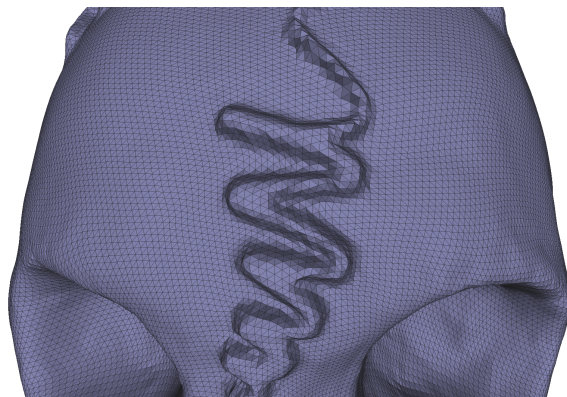
Comme l'ont montré les résultats expérimentaux, notre approche pour le remaillage semi-régulier permet d'obtenir des maillages de bonne qualité en se comparant aux



(a) L'objet SKULL remaillé avec notre schéma adaptatif (140544 triangles).



(b) l'objet SKULL original (irrégulier).



(c) L'objet SKULL remaillé avec [64] (262144 triangles).

FIGURE 2.19 – L'objet SKULL remaillé avec notre schéma adaptatif et avec [64]. Nous remarquons qu'avec notre schéma adaptatif, nous obtenons un maillage avec moins de triangles dans les régions lisses, tout en ayant une meilleure qualité visuelle.

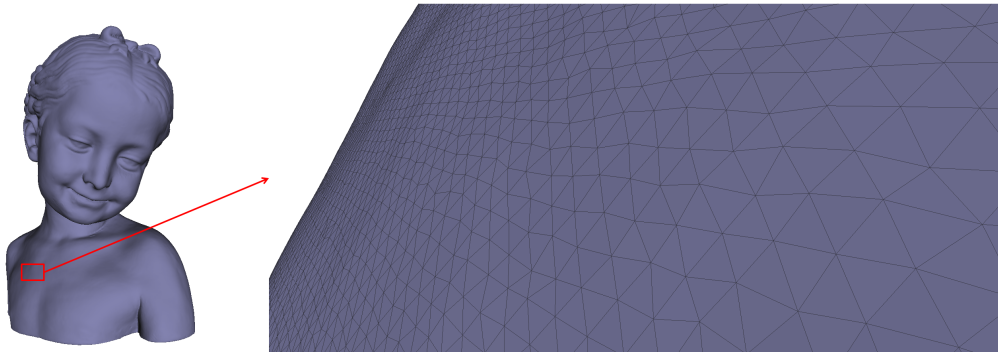


FIGURE 2.20 – L'objet Bimba remaillé avec notre méthode.

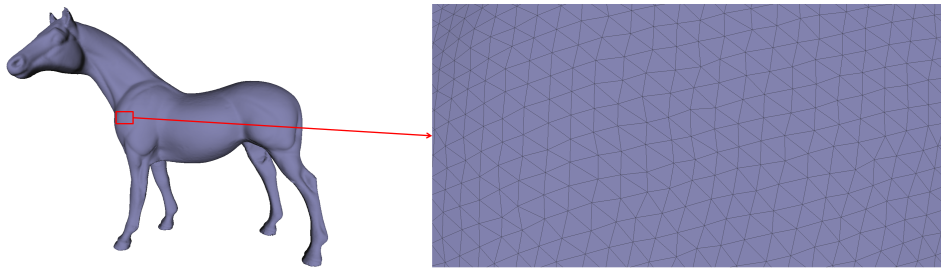


FIGURE 2.21 – L'objet Horse remaillé avec notre méthode.

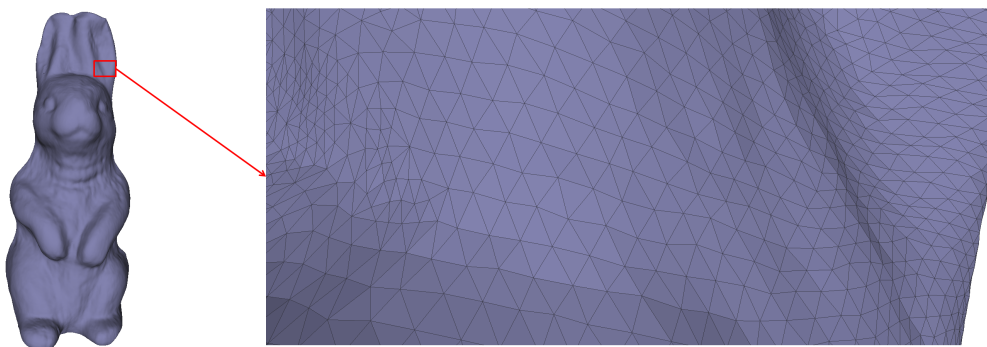


FIGURE 2.22 – L'objet Rabbit remaillé avec notre méthode.

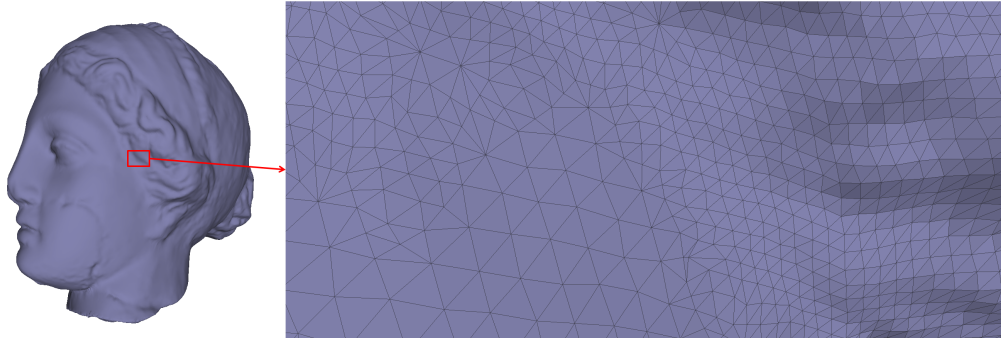


FIGURE 2.23 – L'objet VENUS remaillé avec notre méthode.

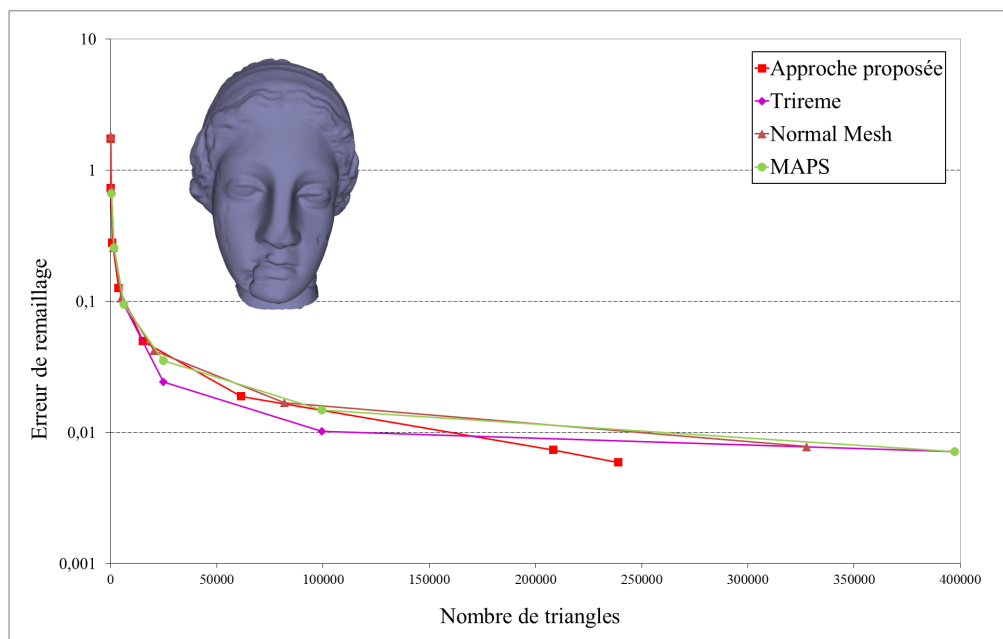


FIGURE 2.24 – Courbes d'évolution de l'erreur de remaillage (échelle logarithmique) pour l'objet VENUS durant l'étape de raffinement.

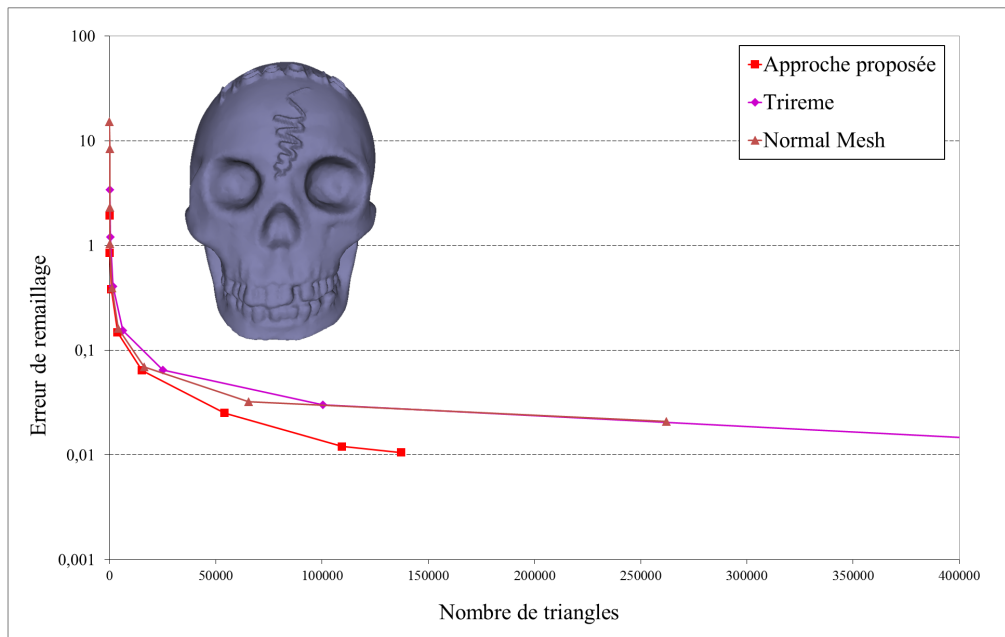


FIGURE 2.25 – Courbes d'évolution de l'erreur de remailage (échelle logarithmique) pour l'objet SKULL durant l'étape de raffinement.

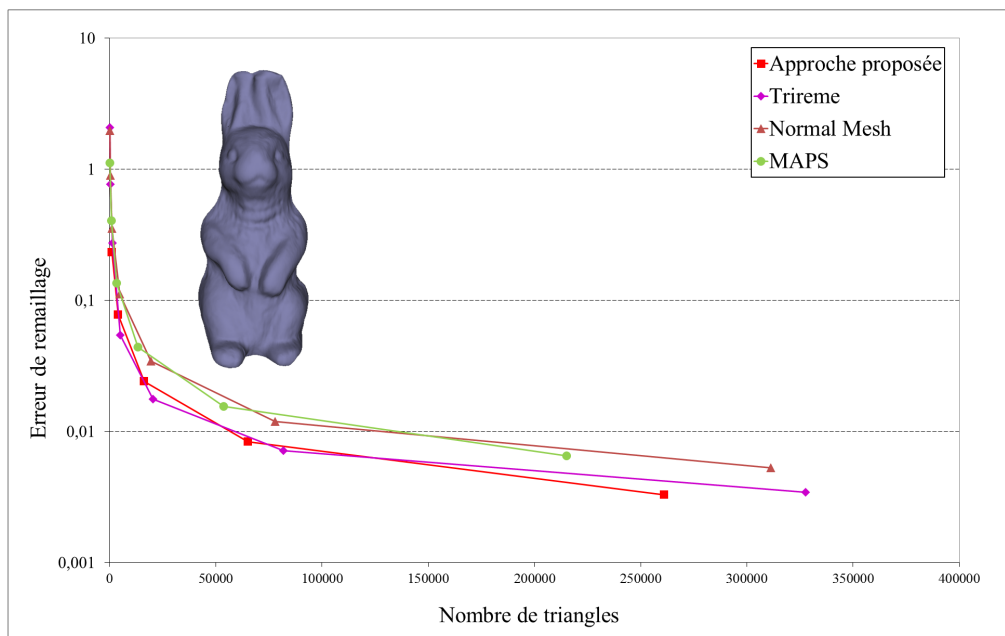


FIGURE 2.26 – Courbes d'évolution de l'erreur de remailage (échelle logarithmique) pour l'objet RABBIT durant l'étape de raffinement.

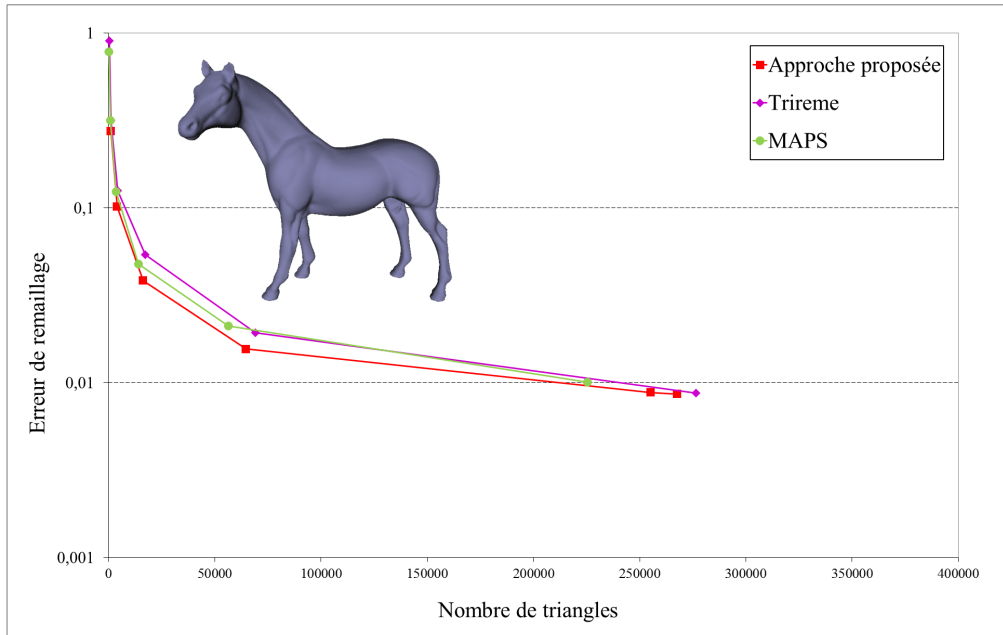


FIGURE 2.27 – Courbes d’évolution de l’erreur de remaillage (échelle logarithmique) pour l’objet HORSE durant l’étape de raffinement.

autres approches tirées de l’état de l’art. L’avantage majeur de notre approche est sa simplicité. En effet, elle est entièrement basée sur la construction d’un diagramme de Voronoï multirésolution sur le maillage irrégulier original. Par suite, elle ne fait pas recours à des calculs complexes qui peuvent nécessiter un noyau de calcul exact pour éliminer des éventuels problèmes d’arrondis.

En plus de sa simplicité, notre approche permet aussi d’avoir une triangulation de bonne qualité. En effet, du fait qu’elle est basée sur les diagrammes de Voronoï, elle produit des maillages avec des triangles qui tendent à être équilatéraux.

Une amélioration possible de notre approche est de mieux gérer les maillages avec des bords ou des objets type CAO (avec des arêtes saillantes). En effet, dans ces deux cas, le maillage semi-régulier obtenu ne préserve pas les bords et les arêtes saillantes. Pour remédier à ce problème, il faudrait initialiser les sites de Voronoï sur les arêtes saillantes ou les bords.

2.6 Conclusion

Dans ce chapitre, nous avons proposé une méthode originale pour le remaillage semi-régulier. Un diagramme de Voronoï multirésolution est construit sur le maillage original, en commençant par construire un diagramme grossier et en lui ajoutant, par la suite, des graines de Voronoï d’une manière adaptative. En utilisant notre méthode, nous avons obtenu des maillages semi-réguliers de haute qualité. Les résultats expérimentaux obte-

nus confirment bien l'intérêt d'une approche adaptative, et basée sur le diagramme de Voronoï.

En comparant les résultats obtenus par notre méthode avec ceux obtenus par des remailleurs publiés dans la littérature, nous pouvons énoncer les remarques suivantes :

- à basse résolution, quand la subdivision est effectuée d'une manière uniforme, notre méthode obtient la plus petite erreur de remailage (à basse résolution, tous les triangles ont une grande masse). Ceci prouve l'intérêt d'utiliser une approche en se basant sur le diagramme de Voronoï ;
- à haute résolution, quand la subdivision est effectuée d'une manière adaptative, nous obtenons de très faibles erreurs de remailage, tout en réduisant le nombre de triangles. Ceci prouve l'intérêt d'utiliser une approche adaptative.

Chapitre 3

Optimisation de deux transformées en ondelettes pour maillages semi-réguliers

3.1 Introduction

Les origines des ondelettes remontent à la fin des années 80 dans la théorie d'approximation [33] et dans le traitement du signal [99]. Depuis, les ondelettes sont les techniques les plus populaires pour représenter les données multirésolutions. Elles ont été utilisées dans un grand nombre de domaines, tels que : la physique, l'analyse de signaux biomédicaux ou encore le traitement d'images. Mais, elles ont été particulièrement utilisées en compression des données. Ainsi, elles permettent une représentation compacte des données et, par conséquent, les taux de compression seront améliorés.

En infographie, la représentation compacte des données n'est pas le seul avantage des ondelettes. En effet, les techniques actuelles d'acquisition haute résolution produisent des maillages surfaciques denses et très détaillés. Notons toutefois que les données monorésolutions massifs sont difficiles à manipuler et à stocker. Ajoutons qu'elles ne sont pas adaptées dans certaines applications, notamment la transmission progressive et rapide dans les applications à bande passante limitée, et l'affichage adaptatif avec plusieurs niveaux de détails. Les ondelettes sont une solution pour surmonter ces difficultés. La structure multirésolution des ondelettes rend le traitement progressif plus facile. La figure 3.1 donne un aperçu d'une décomposition en ondelettes pour les maillages surfaciques.

Contrairement aux autres signaux multimédia, telles que les images et les vidéos, les maillages surfaciques ont souvent un échantillonnage irrégulier, ce qui pose un problème pour y appliquer des transformées en ondelettes. Malgré le développement d'ondelettes pour maillages irréguliers lors de la dernière décennie [63, 136, 18], la solution la plus

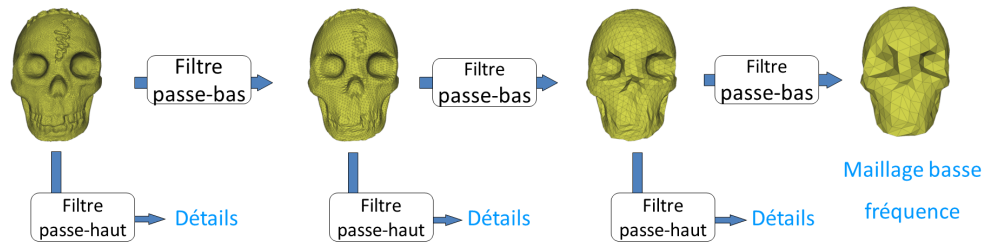


FIGURE 3.1 – Aperçu d’une décomposition en ondelettes pour maillages surfaciques.

efficace, face à cet échantillonnage irrégulier a été l’utilisation des techniques de re-maillage semi-régulier [97, 46, 56, 64, 87, 61, 88, 53, 6, 83]. Même si l’application des ondelettes sur les maillages surfaciques nécessite souvent une étape de remaillage, comme le pré-traitement, nous retrouvons les ondelettes dans de nombreuses applications en infographie, telles que l’édition [128, 146], la segmentation [116], la déformation contrainte par le volume [119] ou encore la compression [79, 35, 112, 113, 86, 29, 124, 77, 31].

Le reste de ce chapitre est organisé de la manière suivante : La section 3.2 donne un état de l’art sur les ondelettes pour les maillages surfaciques. La section 3.3 présente nos motivations et nos contributions. Par la suite, nous détaillons, dans la section 3.4, les notions de bases dont nous aurons besoin. Les sections 3.5 et 3.6 présentent les méthodes que nous avons développées pour optimiser les transformées en ondelettes basées Butterfly et Loop.

3.2 Etat de l’art

Dans cette section, nous présentons une synthèse sur les travaux développés dans la littérature concernant les ondelettes et la compression basée ondelettes pour les maillages semi-réguliers.

3.2.1 Ondelettes pour les maillages semi-réguliers

Les chercheurs Lounsbery *et al.* sont considérés comme les pionniers dans le développement des ondelettes pour les maillages surfaciques à topologie arbitraire [98]. En effet, ils ont proposé une technique pour construire des transformées en ondelettes à partir de n’importe quel schéma de subdivision, qui est local, stationnaire, continu et uniformément convergent, comme les schémas de Catmull-Clark [25], Loop [94] et Butterfly [45]. Le schéma de subdivision représente alors le filtre de synthèse et le filtre d’analyse en est dérivé. Deux filtres sont finalement appliqués sur le maillage d’entrée en cours d’analyse, qui permettent de fournir un maillage de faible résolution (filtrage passe-bas) et un ensemble de coefficients d’ondelettes (filtrage passe-haut).

Schröder et Sweldens ont développé une méthode permettant de construire des transformées en ondelettes pour des fonctions scalaires, qui peuvent être définies sur une sphère [121], et ce, en se basant sur les travaux de Lounsbery *et al.* [98] et ceux de Donoho [40], concernant les transformées en ondelettes interpolantes. Cependant, ils n'étaient pas les premiers à avoir proposé de telles transformées. En effet, les premiers sont Dahlke *et al.*, qui ont utilisé une base de produit tensoriel, où un élément représente une spline exponentielle [32]. En outre, Freedman et Windheuser ont proposé une transformée continue, où sa forme semi-discrète est développée dans [52]. Néanmoins, les travaux de Schröder et Sweldens sont remarquables parce qu'ils étaient les premiers à montrer, d'une part l'efficacité du schéma *lifting* [129, 130] pour la construction des transformées en ondelettes biorthogonales [121], et d'autre part la simplicité d'implémentation des filtres résultants (qui nécessitent des calculs locaux), en utilisant ce schéma. Par ailleurs, cette technique est indépendante de la paramétrisation de la surface. Kovacevic et Sweldens ont par la suite généralisé le concept de la transformée en ondelettes pour toute dimension de maillages [84]. Ils ont montré que le schéma *lifting* permet de construire des bancs de filtres et des transformées en ondelettes pour tout type de maillage, qui peut être de grande dimension et ayant un nombre important de moments (primaux/duaux) nuls. Ajoutons qu'ils ont également montré que seulement deux étapes sont nécessaires (prédiction et mise à jour) pour une transformée en ondelettes, à condition que les fonctions d'échelle associées soient interpolantes.

En 2000, Khodakovsky *et al.* ont proposé une transformée en ondelettes pour les maillages semi-réguliers, qui est basée sur le schéma approximant de Loop [79]. Cependant, les premiers travaux développés sur les transformées en ondelettes pour les maillages semi-réguliers ont été basés sur des schémas de subdivision interpolants. Le schéma de subdivision de Loop est utilisé durant la synthèse, comme un filtre de reconstruction passe-bas, tandis que le filtre passe-haut en est dérivé pour former avec le filtre passe-bas des filtres miroirs en quadrature. L'inconvénient de cette approche est que les filtres du schéma de subdivision de Loop ne peuvent pas être appliqués directement. En effet, contrairement aux transformées basées sur le schéma *lifting*, les coefficients d'ondelettes et le maillage basse résolution sont obtenus en résolvant un système linéaire parcimonieux, qui se base sur les filtres de reconstruction passe-bas et passe-haut. Récemment, Bertram [15] a résolu le problème en proposant une transformée en ondelettes biorthogonales, qui est basée sur le schéma de Loop et le schéma *lifting*. En 2003, Li *et al.* [90] ont également proposé une transformée en ondelettes, en se basant uniquement sur le schéma de Loop.

3.2.2 Compression basé-ondelettes des maillages semi-réguliers

La compression permet un stockage compact et une transmission rapide dans les applications à bande passante limitée des maillages massives. De nombreuses techniques de compression de maillages 3D ont été déjà proposées [68]. A notre connaissance, les codeurs basés ondelettes qui prennent en entrée des maillages semi-réguliers

[79, 35, 112, 113, 86, 29, 124, 77, 31] sont les plus efficaces. En effet, la régularité d'échantillonnage de ces maillages semi-réguliers permet l'application efficace d'une transformée en ondelettes. Dans ce qui suit, nous présentons brièvement les principaux travaux dans ce domaine.

Le premier codeur basé ondelettes (souvent appelé PGC) pour les maillages semi-réguliers a été proposé par Khodakovsky *et al.* [79]. Ce codeur est basé sur des *quadtree* multi-échelles et la scalabilité de la qualité. Les auteurs ont proposé une transformée en ondelettes basée Loop, mais n'importe quelle autre transformée en ondelettes peut être utilisée. Un codeur *zerotree* suivi d'un codage entropique sont appliqués en parallèle sur chaque composante (tangentielle et normale) des coefficients d'ondelettes calculés dans un repère local. Ce codeur a été également proposé pour les maillages "normaux" [77]. La seule différence est le choix de la transformée en ondelettes. En effet, pour ces maillages "normaux", les auteurs utilisent la transformée en ondelettes non-liftée basée Butterfly, qui est optimale pour ce type de maillage (lors de la construction d'un maillage "normal", un niveau de résolution est obtenu en subdivisant le niveau précédent à l'aide du masque de subdivision du schéma de Butterfly).

Ensuite, plusieurs techniques d'allocation binaire ont été proposées pour améliorer les performances de ce codeur [112, 113, 124, 86]. L'idée est d'inclure un processus d'allocation débit lors de l'étape de quantification afin d'optimiser le compromis débit-distorsion, en d'autres termes, d'atteindre la qualité maximale pour un débit fixe (ou vice-versa).

Récemment, Denis *et al.* ont proposé un codeur supportant la scalabilité de la qualité [35]. Ce codeur exploite les dépendances intra-bandes (ou les dépendances statistiques entre les coefficients d'ondelettes). Les sous-bandes de coefficients d'ondelettes sont alors codées indépendamment à l'aide d'un codeur basé *octree* et un codage entropique contextuel. Les expériences expérimentaux ont montré que ce codeur fournit de meilleurs résultats que le codeur PGC et des résultats similaires avec le codeur proposé dans [112], qui ne supporte pas la scalabilité de la qualité.

3.3 Motivation et contributions

En compression d'images, il existe des travaux qui améliorent les transformées en ondelettes en adaptant les filtres de prédiction et de mise à jour du schéma *lifting* aux caractéristiques de l'image traitée [14, 115]. Inspirés par cette idée de schéma *lifting* adaptatif, nous proposons d'optimiser deux transformées en ondelettes pour les maillages semi-réguliers. En effet, les transformées en ondelettes ont des filtres fixes ne dépendant pas de la géométrie de l'objet, ce qui réduit leur efficacité dans certains cas. Les transformées en ondelettes basées Butterfly et Loop sont les deux transformées les plus utilisées pour les maillages semi-réguliers.

Le schéma de subdivision de Butterfly est un schéma interpolant et, par conséquent, il peut générer des surfaces qui ne sont pas lisses, en essayant d'interpoler tous les points.

Ceci peut se traduire lors de la transformée en ondelettes par des coefficients en ondelettes de grandes amplitudes. La figure 3.2 montre un exemple où la subdivision avec le schéma de Butterfly n'est pas bien adaptée au maillage original.

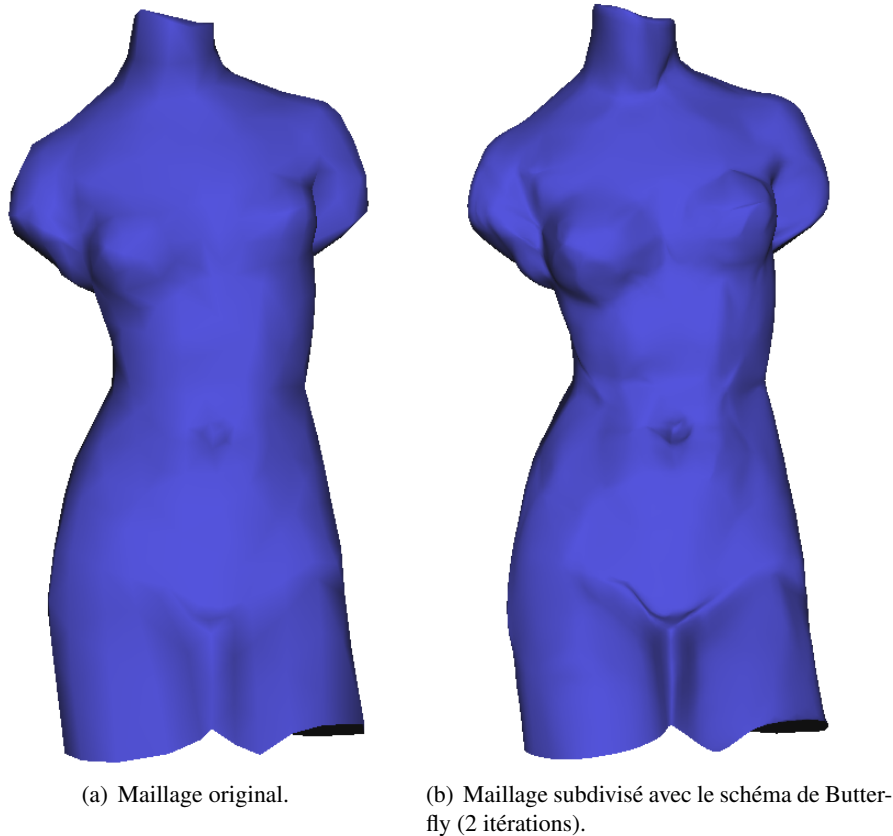


FIGURE 3.2 – Exemple où la subdivision avec le schéma de Butterfly n'est pas bien adaptée au maillage original. Avec le schéma de Butterfly on a perdu l'aspect lisse de la surface originale.

Le schéma de subdivision de Loop est un schéma approximant et, par conséquent, il a tendance à lisser les surfaces rugueuses (qui contiennent un grand nombre de détails hautes fréquences). Ceci se traduit lors de la transformée en ondelettes par des coefficients en ondelettes de grandes amplitudes. La figure 3.3 montre un exemple où le schéma de Loop n'est pas bien adaptée au maillage original.

Il serait donc préférable d'avoir des transformées en ondelettes qui s'adaptent en fonction des caractéristiques géométriques de l'objet. Ceci pourrait être possible, en adaptant les filtres de prédiction et de mise à jour (qui sont décrits dans [115]) de ces deux transformées implémentées en schéma *lifting*. Ainsi, nous proposons ici un algorithme pour adapter ces deux transformées en ondelettes pour maillages semi-réguliers. Notre motivation est d'améliorer les performances des codeurs basés ondelettes existants et la qualité des maillages obtenus aux basses résolutions pendant l'analyse. Nous

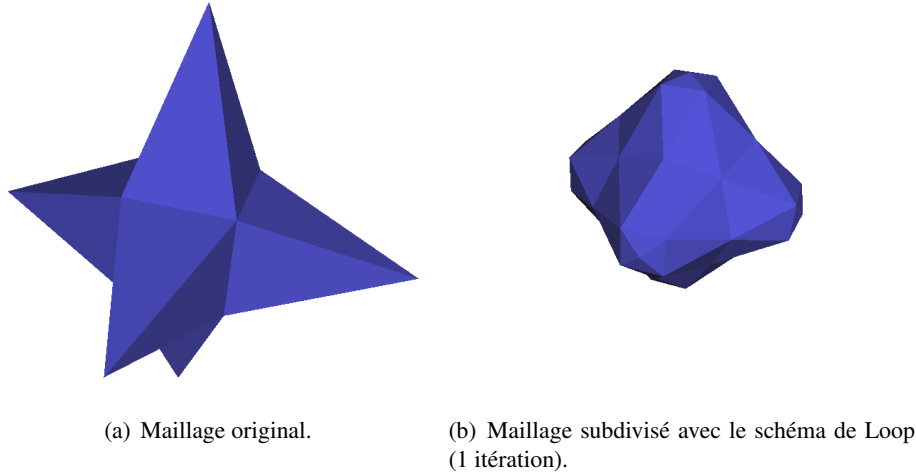


FIGURE 3.3 – Exemple où la subdivision avec le schéma de Loop n’est pas bien adaptée au maillage original. Avec le schéma de Loop on a perdu les détails hautes fréquences (les pointes de l’étoile).

proposons donc d’optimiser les filtres des différentes étapes (prédiction et mise à jour) pendant l’analyse, en vue de maximiser la parcimonie des coefficients d’ondelettes à chaque niveau de résolution.

3.4 Notions de base

Cette section présente les notions de base sur la transformée en ondelettes pour maillages semi-réguliers, et plus particulièrement le schéma *lifting*.

3.4.1 Transformée en ondelettes pour les maillages semi-réguliers

Nous noterons par V^j l’ensemble des sommets du maillage semi-régulier à une résolution donnée M^j , où chaque sommet est défini par sa position dans l’espace Euclidien. En appliquant une transformée en ondelettes à M^j , nous obtenons un maillage de résolution inférieure M^{j-1} , qui est défini par un ensemble de sommets V^{j-1} , et un ensemble de coefficients d’ondelettes C^j (vecteurs 3D). La figure 3.4 illustre une transformée en ondelettes d’un maillage semi-régulier. Les coefficients sont représentés dans un ensemble de repères locaux en 3D, qui sont induits par le plan tangent et la normale à la surface [146]. Dans un schéma de codage basé ondelettes, la transformée illustrée figure 3.4 représente l’étape d’analyse. En inversant les filtres utilisés lors de l’analyse, il est possible de retrouver M^j à partir de C^j et M^{j-1} . Cette étape de reconstruction est appelée synthèse.

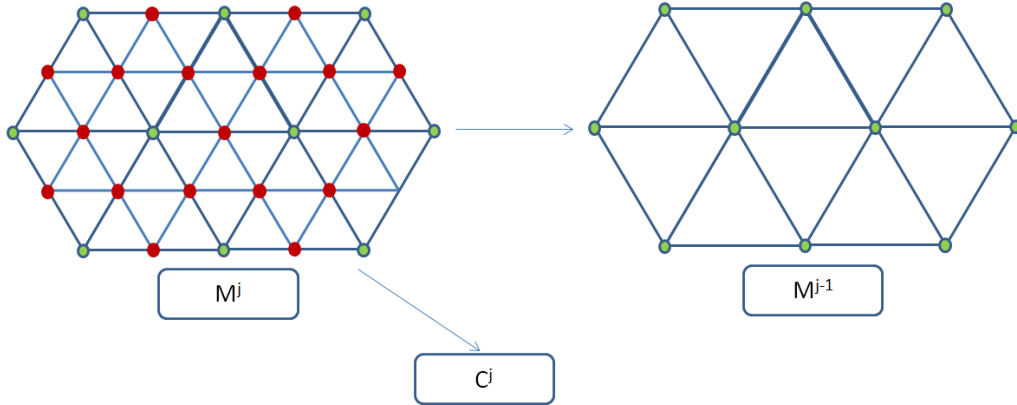


FIGURE 3.4 – Transformée en ondelettes d'un maillage semi-régulier : M^j étant transformé en un maillage de résolution inférieure M^{j-1} et un ensemble de coefficients d'ondelettes C^j (associés aux sommets rouges supprimés lors de la transformée).

3.4.2 Schéma *lifting*

Le schéma *lifting* est un outil efficace pour le calcul des transformées en ondelettes [129, 130]. La principale caractéristique du schéma *lifting* est que toutes les constructions sont faites dans le domaine spatial, alors que l'approche traditionnelle repose sur le domaine fréquentiel. Le schéma *lifting* a les avantages suivants [131] :

- il ne nécessite pas l'analyse de Fourier ;
- il permet d'obtenir des algorithmes pouvant être généralisés à des situations géométriques complexes, ce qui se produit souvent en infographie ;
- il permet une transformée *in-situ*, réduisant ainsi la quantité de mémoire nécessaire ;
- il permet d'obtenir une implémentation réversible (analyse / synthèse) plus rapide qu'une transformée implémentée avec des bancs de filtres.

Le schéma *lifting*, tel que donné figure 3.5, est basé sur les trois étapes suivantes :

- *split*, où les données en entrée sont divisées en deux sous-ensembles. Dans notre contexte, les sommets V^j sont divisés en deux sous-ensembles V_0^j et V_1^j (sommets verts et rouges de la figure 3.4) ;
- *prédiction*, où les positions des sommets V_1^j (points rouges) sont prédites à partir des sommets V_0^j (points verts), en utilisant un filtre de prédiction P , tel que : $V_1^j = P(V_0^j)$. Les erreurs de prédiction correspondent aux coefficients d'ondelettes C^j ;
- *mise à jour*, dans laquelle les positions des sommets V_0^j sont modifiées à partir des coefficients d'ondelettes déjà calculés C^j , en utilisant un filtre de mise à jour U . Un gain ($\times 2$) est appliqué aux sommets modifiés pour obtenir le maillage M^{j-1} , qui est défini à partir de l'ensemble de sommets V^{j-1} .

Pour la synthèse, il suffit de renverser l'ordre et le signe des différentes étapes (voir figure 3.6).

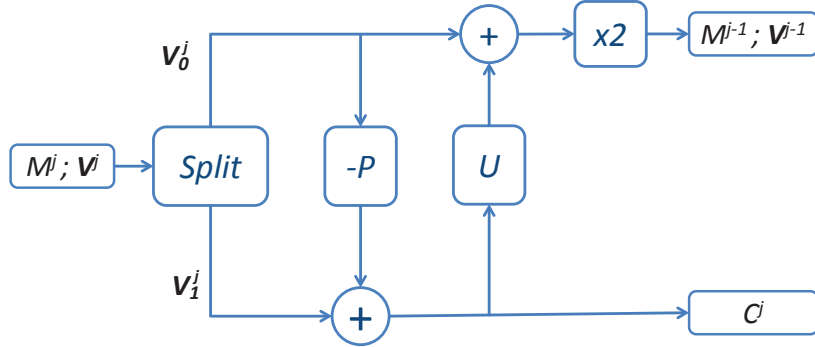


FIGURE 3.5 – Schéma *lifting* pour les maillages semi-réguliers : Analyse.

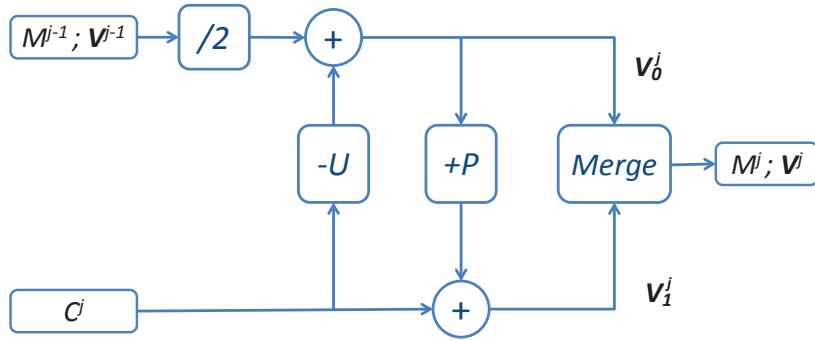


FIGURE 3.6 – Schéma *lifting* pour les maillages semi-réguliers : Synthèse.

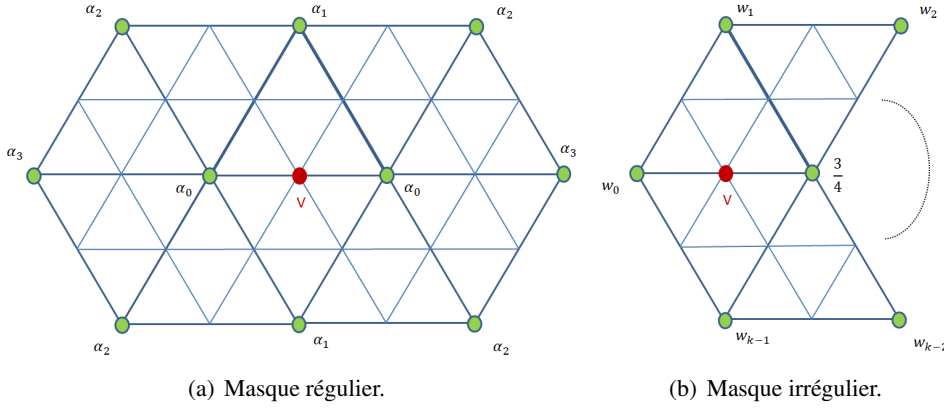
3.5 Schéma *lifting* adaptatif basé Butterfly

Dans cette section, nous présentons notre contribution en adaptant le schéma *lifting* basé Butterfly. L'objectif visé est d'optimiser les filtres de prédiction et de mise à jour de ce schéma afin d'améliorer l'efficacité de la quantification ultérieure. Notre démarche d'optimisation consiste à calculer de nouvelles pondérations pour les deux filtres P et U . Cette optimisation se fait à chaque niveau de décomposition durant l'analyse. Nous obtenons donc un ensemble de filtres P et U par niveau de résolution.

3.5.1 Optimisation de l'étape de prédiction

Pour l'étape de prédiction, nous partons du schéma de Butterfly modifié, qui est proposé dans [145]. Ce schéma permet de prédire la position d'un sommet v_1^j appartenant à V_1^j à partir de ses sommets voisins dans V_0^j . Le voisinage est spécifié par l'un des deux masques suivants (régulier ou irrégulier), qui sont donnés figure 3.7 :

- le masque régulier (figure 3.7(a)), utilisé quand, pour un sommet donné de V_1^j , ses deux voisins directs de V_0^j sont réguliers (valence 6) ;
- le masque irrégulier, utilisé pour tout autre cas de figure, autrement dit au moins un des deux voisins directs de V_0^j est irrégulier (figure 3.7(b)).

FIGURE 3.7 – Masques utilisés lors de la prédiction du schéma *lifting* basé Butterfly.

Il convient de noter ici que nous n’optimisons pas le cas irrégulier, car il y a beaucoup de configurations possibles pour ce filtre, et ce, en fonction de la valence des deux voisins directs. Le fait d’optimiser tous les masques irréguliers nécessiterait le calcul et la transmission d’informations secondaires trop grandes pour pouvoir reconstituer les maillages pendant la synthèse. Au lieu de cela, nous avons préféré optimiser uniquement le cas régulier et utiliser les poids originaux $\{w_i\}$, qui sont donnés dans [145], pour le cas irrégulier.

En considérant un niveau de résolution j , nous pouvons calculer l’ensemble des coefficients d’ondelettes C^j à partir de l’équation suivante :

$$\mathbf{C}^j = V_1^j - P(V_0^j). \quad (3.1)$$

Comme pour la majorité des codeurs géométriques basés ondelettes, qui concernent les maillages semi-réguliers, nous considérons séparément les composantes tangentielles (correspondant aux coordonnées x et y des coefficients exprimés dans un repère local) et les composantes normales (correspondant aux coordonnées z exprimées dans le même repère local). Nous proposons donc de calculer deux filtres de prédiction optimisés. Il s’agit du filtre P_{xy} , qui permet de calculer des composantes tangentielles, et du filtre P_z permettant le calcul des composantes normales. Le même algorithme est donc utilisé pour l’optimisation de ces deux filtres. Par suite, et pour plus de clarté, nous utiliserons P pour les deux filtres P_{xy} et P_z .

Afin d’augmenter les taux de compression, nous proposons de rendre l’ensemble des coefficients C^j aussi parcimonieux que possible. Pour cela, et en s’inspirant de travaux précédents [39] que l’on a décidé de minimiser la norme L_1 de l’ensemble des coefficients, et ce, à chaque niveau de résolution. En effet, le fait de maximiser la parcimonie des coefficients revient à résoudre le problème de minimisation suivant :

$$\min_{\alpha^j} \|V_1^j - P_{\alpha^j}(V_0^j)\|_1, \quad (3.2)$$

où α^j définit les 10 poids relatifs au filtre P pour le niveau de résolution j (voir figure 3.7(a)).

Le problème de minimisation posé (3.2) doit être résolu durant l'analyse. De ce fait, nous le résolvons à chaque niveau de résolution, en commençant par le niveau le plus fin.

Pour un niveau de résolution j donné, nous pouvons définir le vecteur des inconnus x qui contient les 10 poids de P , soit :

$$x = (\alpha_0^j, \alpha_1^j, \dots, \alpha_9^j)^T. \quad (3.3)$$

Nous noterons par $\mathcal{N}(\cdot)$ l'ensemble des 10 sommets voisins appartenant à V_0^j et dépendant du masque régulier. Considérons une matrice A de dimension $(n_r \times 10)$, telle que :

$$A = [\mathcal{N}(v_1^j(0)); \mathcal{N}(v_1^j(1)); \dots; \mathcal{N}(v_1^j(n_r - 1))], \quad (3.4)$$

où n_r est le nombre de sommets de V_1^j , sur lequel le masque régulier est appliqué. De même, considérons un vecteur b , tel que :

$$b = (v_1^j(0), v_1^j(1), \dots, v_1^j(n_r - 1))^T. \quad (3.5)$$

La résolution du problème posé (3.2) peut être menée à partir de la minimisation de la fonction suivante :

$$\begin{aligned} f : \mathbb{R}^{10} &\mapsto \mathbb{R} \\ \mathbf{x} &\mapsto f(\mathbf{x}) = \|A\mathbf{x} - \mathbf{b}\|_1. \end{aligned} \quad (3.6)$$

Du fait que le filtre de prédiction proposé dans [145] est symétrique, alors nous pouvons écrire les conditions suivantes :

$$\begin{aligned} \alpha_0 &= \alpha_1 \\ \alpha_2 &= \alpha_3 \\ \alpha_4 &= \alpha_5 = \alpha_6 = \alpha_7 \\ \alpha_8 &= \alpha_9. \end{aligned} \quad (3.7)$$

Ainsi, nous pouvons récrire la fonction f (qui n'a désormais que quatre variables α_0 , α_2 , α_4 et α_8) à minimiser de l'équation (3.6) sous la forme suivante :

$$\begin{aligned} f : \mathbb{R}^4 &\mapsto \mathbb{R} \\ \mathbf{x} &\mapsto f(\mathbf{x}) = \|A\mathbf{x} - \mathbf{b}\|_1. \end{aligned} \quad (3.8)$$

Preuve de convexité Nous pouvons écrire la fonction f à minimiser de la manière suivante :

$$\begin{aligned}
 f(\mathbf{x}) &= \sum_{i=0}^{n_r-1} |A_i \mathbf{x} - b_i|, \\
 &= \sum_{i=0}^{n_r-1} g_i(\mathbf{x}),
 \end{aligned}
 \tag{3.9}$$

$$\tag{3.10}$$

où A_i sont les lignes de A et b_i sont les valeurs de \mathbf{b} .

Soient $(x, y) \in \mathbb{R}^{10}$ et $\lambda \in [0, 1]$. Nous pouvons écrire :

$$\begin{aligned}
 g_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) &= |A_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) - b_i|, \\
 &= |\lambda A_i \mathbf{x} + (1 - \lambda) A_i \mathbf{y} - b_i|, \\
 &= |\lambda(A_i \mathbf{x} - b_i) + (1 - \lambda)(A_i \mathbf{y} - b_i)|, \\
 &\leq |\lambda(A_i \mathbf{x} - b_i)| + |(1 - \lambda)(A_i \mathbf{y} - b_i)|.
 \end{aligned}
 \tag{3.11}$$

Sachant que $\lambda \in [0, 1]$, il en résulte donc que λ et $(1 - \lambda)$ sont positifs. Ainsi, nous pouvons écrire l'inégalité suivante :

$$g_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda |A_i \mathbf{x} - b_i| + (1 - \lambda) |A_i \mathbf{y} - b_i|,$$

qui s'écrit

$$g_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda g_i(\mathbf{x}) + (1 - \lambda) g_i(\mathbf{y}).$$

Ce qui prouve que les fonctions g_i sont convexes. Ainsi, la fonction f qui est une somme de ces fonctions g_i , est par conséquent convexe. En plus, la fonction f , qui est définie positive, admet un minimum unique global. Pour déterminer ce minimum, nous avons utilisé l'algorithme de simplex de Nelder-Mead [85] (voir annexe 6.6) qui est un algorithme qui nécessite pas de calculer les dérivées de la fonction et qui est efficace pour les fonctions non-dérivables (ce qui est le cas de la fonction qu'on veut minimiser). Notons toutefois que d'autres types d'algorithmes peuvent être utilisés, en vue de déterminer ce minimum.

3.5.2 Calcul du nouveau filtre de mise à jour

Nous considérons ici le même niveau de résolution j et le même schéma *lifting*, qui est représenté figure 3.5. L'ensemble des sommets \mathbf{V}^{j-1} du niveau de résolution $j - 1$ est mis à jour une fois que l'étape de prédiction est terminée, et ce, en utilisant l'équation suivante :

$$V^{j-1} = 2 \times (V_0^j + U^j(C^j)), \tag{3.12}$$

où U^j est le filtre de mise à jour. Ce filtre, qui dépend de la valeur d'un paramètre γ , est associé au masque illustré figure 3.8.

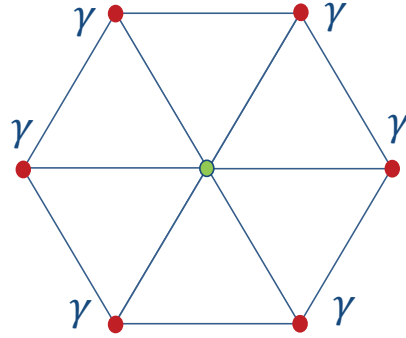


FIGURE 3.8 – Masque utilisé lors de la mise à jour du schéma *lifting* basé Butterfly.

Quand la prédiction P est basée sur un schéma de subdivision interpolant, le filtre de mise à jour U est choisi pour préserver certains moments des données traitées [131]. La modification du filtre P nous amène donc à modifier le filtre U , et par conséquent, la valeur du paramètre γ .

Nous calculons la valeur du paramètre γ afin de préserver la moyenne (le moment d'ordre 0) entre V^j et V^{j-1} . Pour cela, nous avons utilisé la méthode proposée dans [131]. L'idée est de mettre tous les sommets de V^{j-1} et tous les coefficients de C^j à zéro, sauf pour un coefficient qui sera mis à 1 (voir figure 3.9).

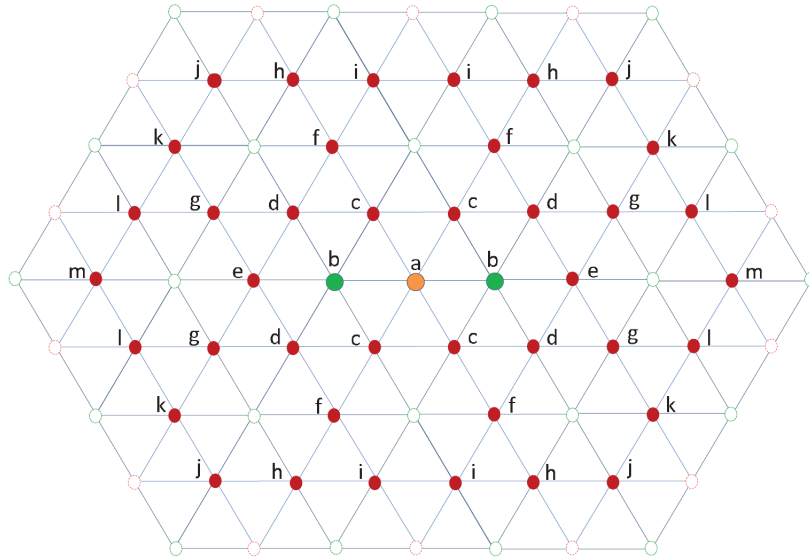


FIGURE 3.9 – Méthode de calcul de la valeur du paramètre γ : le point orange (a) représente le seul coefficient non nul. Les points verts (b) représentent les deux seuls sommets de V^{j-1} qui seront non nuls après une étape de synthèse dans ces conditions. Les points rouges représentent les 43 sommets de V^j qui seront non nuls après une étape de synthèse dans ces conditions.

Ensuite, nous appliquons les filtres de synthèse, en vue d'obtenir V^j . La majorité

des sommets, ainsi obtenus, ont des coordonnées nulles, sauf pour 43 d'entre eux. Ces sommets sont illustrés figure 3.9 et le tableau 3.1 donne les valeurs de ces sommets.

TABLE 3.1 – Valeurs des 43 sommets non nuls de \mathbf{V}^j obtenus après une étape de synthèse sur un maillage \mathbf{V}^{j-1} composé uniquement de sommets nuls et un ensemble de coefficients \mathbf{C}^j contenant un seul élément non nul égal à 1.

Index	Nombre	Valeur
a	1	$1 - 2\gamma\alpha_0$
b	2	$-\gamma$
c	4	$-\gamma(\alpha_0 + \alpha_2)$
d	4	$-\gamma(\alpha_0 + \alpha_4)$
e	2	$-\gamma(\alpha_0 + \alpha_8)$
f	4	$-\gamma(\alpha_2 + \alpha_4)$
g	4	$-\gamma\alpha_2$
h	4	$-\gamma\alpha_4$
i	4	$-\gamma(\alpha_4 + \alpha_8)$
j	4	$-\gamma\alpha_8$
k	4	$-\gamma\alpha_4$
l	4	$-\gamma\alpha_4$
m	2	$-\gamma\alpha_8$

Dans ce cas, la moyenne de l'ensemble des sommets \mathbf{V}^j peut être déterminée à partir de l'expression suivante :

$$\frac{1}{43} (1 - 2\gamma - 12\gamma\alpha_0 - 12\gamma\alpha_2 - 24\gamma\alpha_4 - 12\gamma\alpha_8). \quad (3.13)$$

La moyenne de l'ensemble des sommets \mathbf{V}^{j-1} étant nulle et la moyenne de l'ensemble des sommets \mathbf{V}^j , tel que définie par (3.13), doit être égale à zéro. Nous obtenons finalement l'expression suivante du paramètre λ :

$$\gamma = \frac{1}{2 + 12\alpha_0 + 12\alpha_2 + 24\alpha_4 + 12\alpha_8}, \quad (3.14)$$

qui dépend donc des paramètres α_0 , α_2 , α_4 et α_8 , qui ont été évalués précédemment pour optimiser la prédiction.

3.5.3 Validation de la méthode proposée

Cette sous-section est dédiée à la validation de la méthode proposée. Pour cela, nous avons expérimenté notre algorithme, en comparant la norme $L1$ des ensembles des coefficients d'ondelettes obtenus avec ou sans l'algorithme d'optimisation proposé pour les objets VASE LION, BIMBA et JOAN OF ARC. Les tableaux 3.2, 3.3 et 3.4 donnent les résultats obtenus. La colonne "Classique" donne les valeurs de la norme $L1$ des composantes tangentielles (C.T.) et de la composante normale (C.N.) des coefficients d'ondelettes à chaque niveau de résolution, en utilisant le schéma classique basé Butterfly. La colonne

"Méthode OaC" (Optimisation avec Contrainte) montre les mêmes valeurs lorsque la transformée adaptative proposée est appliquée. Nous avons testé également une variante de notre méthode, que nous avons appelé méthode OsC (Optimisation sans Contrainte). Dans cette variante, nous avons enlevé la contrainte sur la symétrie des filtres de prédiction qui a été donnée dans (3.7).

TABLE 3.2 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielle ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet VASE LION, avec ou sans l'algorithme d'optimisation proposé ; OaC et OsC sont deux variantes de l'algorithme proposé (avec ou sans la contrainte sur la symétrie du filtre de prédiction) et res est le niveau de résolution.

res	Classique		Méthode OaC		Méthode OsC	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	83,76	132,21	84,41	126,40	83,80	125,18
2	62,32	134,32	62,21	125,53	62,10	125,50
3	45,80	126,92	45,10	116,87	45,08	116,88
4	31,53	77,88	29,94	69,53	29,93	69,51
5	21,08	42,15	20,22	39,22	20,22	39,21
6	13,81	26,10	13,50	25,78	13,50	25,78

TABLE 3.3 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielle ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet JOAN OF ARC, avec ou sans l'algorithme d'optimisation proposé ; OaC et OsC sont deux variantes de l'algorithme proposé (avec ou sans la contrainte sur la symétrie du filtre de prédiction) et res est le niveau de résolution.

res	Classique		Méthode OaC		Méthode OsC	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	77,25	95,24	78,16	91,77	78,13	91,74
2	38,33	55,13	36,52	51,58	36,46	51,58
3	23,07	30,50	22,43	28,65	22,40	28,65
4	14,39	16,90	14,02	16,28	14,02	16,28
5	9,72	10,88	9,55	10,80	9,55	10,80

Il est important d'indiquer que l'utilisation de la méthode d'optimisation proposée (quelque soit la variante utilisée : OsC ou OaC) permet d'avoir une norme $L1$ des coefficients d'ondelettes plus petite quelle que soit la résolution et la composante, sauf pour la composante tangentielle de la plus basse fréquence. Nous justifions cela par le fait qu'une grande proportion des sommets à cette résolution sont irréguliers. Puisque nous ne pouvons pas adapter le filtre utilisé pour ce genre de sommet, notre algorithme d'optimisation peut être moins efficace à ce niveau de résolution, en raison du nombre trop petit de sommets réguliers. Cette minimisation de la norme $L1$ des coefficients obtenus permettra d'améliorer les taux de compression, comme nous allons le voir dans la

TABLE 3.4 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielle ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet Bimba, avec ou sans l'algorithme d'optimisation proposé ; OaC et OsC sont deux variantes de l'algorithme proposé (avec ou sans la contrainte sur la symétrie du filtre de prédiction) et res est le niveau de résolution.

res	Classique		Méthode OaC		Méthode OsC	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	86,34	148,90	86,69	142,28	86,86	140,53
2	45,17	118,31	40,33	114,37	40,23	114,26
3	25,92	84,29	24,19	80,48	24,17	80,46
4	15,92	54,94	14,65	52,40	14,63	52,37
5	7,29	24,52	6,86	23,14	6,86	23,14
6	4,57	13,33	4,42	12,99	4,42	12,99
7	3,72	12,36	3,64	12,17	3,64	12,17

section 3.7.

3.6 Schéma *lifting* adaptatif basé Loop

Nous avons aussi développé une approche adaptative de la transformée en ondelettes basée Loop, qui a été proposée par Bertram [15]. Cette transformée, qui est représentée figure 3.10, est différente du schéma *lifting* présenté dans la sous-section 3.4.2.

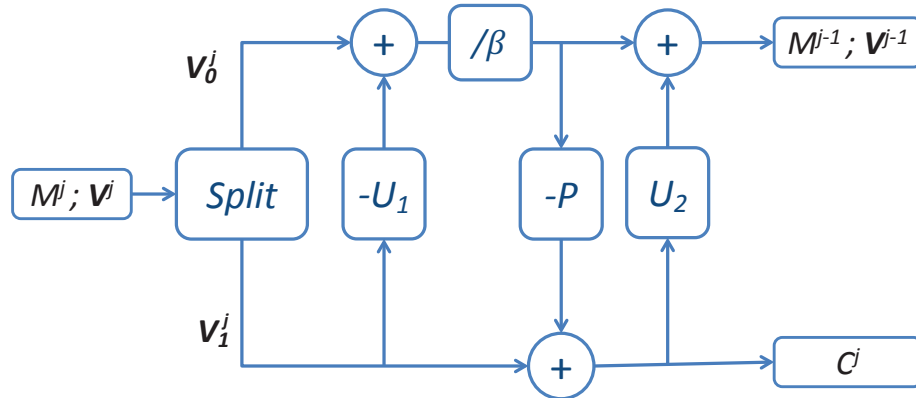


FIGURE 3.10 – Schéma *lifting* basé Loop proposé dans [15].

Une fois le maillage en entrée V^j est divisé en deux sous ensembles V_0^j et V_1^j , le schéma basé Loop, qui est donné dans [15], se résume par les trois étapes majeures suivantes :

- un filtre de prédiction P , qui dépend de deux poids (ρ_0 et ρ_1) et ayant la forme donnée figure 3.11 ;

- un filtre de mise à jour U_1 , qui est appliqué avant l'étape de prédiction. Ce filtre, qui dépend du poids δ , a la forme donnée figure 3.12(a). Un gain $\frac{1}{\beta}$ (qui dépend des valences des sommets) est aussi appliqué après cette étape de mise à jour. Ces deux étapes sont appliquées afin d'obtenir un schéma de subdivision lisse, quand le filtre de prédiction P est utilisé ;
- un deuxième filtre de mise à jour U_2 , qui est appliqué après le premier, dépend des poids ω , tout en ayant la forme donnée figure 3.12(b). Cette étape est appliquée afin d'obtenir une transformée en ondelettes biorthogonale [15].

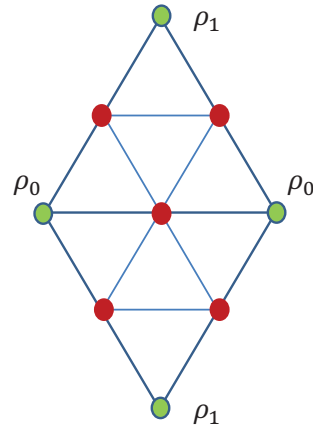
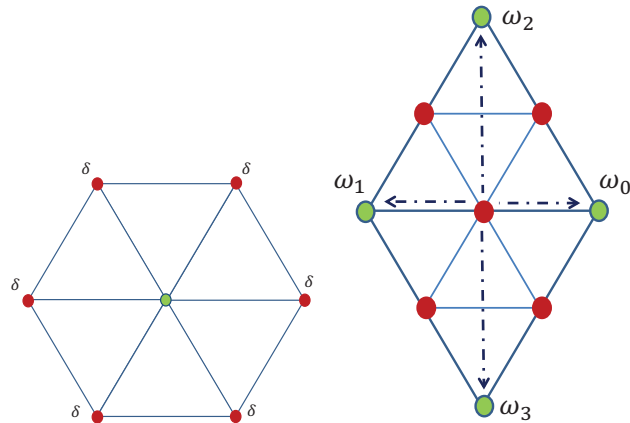


FIGURE 3.11 – Masque utilisé lors de l'étape de prédiction du schéma *lifting* basé Loop.



(a) Masque du premier filtre de mise à jour. (b) Masque du deuxième filtre de mise à jour.

FIGURE 3.12 – Masques utilisés lors de la mise à jour du schéma *lifting* basé Loop.

3.6.1 Optimisation de la prédiction et du premier filtre de mise à jour

Dans cette sous-section, nous allons minimiser la norme L_1 des ensembles de coefficients d'ondelettes à chaque niveau de résolution, comme nous l'avons déjà proposé pour le schéma *lifting* basé Butterfly. Contrairement à ce que nous avons proposé pour le schéma de Butterfly, nous utiliserons le même masque pour les composantes tangentielle et normale afin de rendre possible l'orthogonalisation de la fonction d'ondelette par le filtre de mise à jour U_2 (voir [15]). Soulignons que, si nous utilisons deux filtres différents (pour les composantes tangentielle et normale), alors nous obtenons deux ensembles de fonctions d'ondelettes, et le filtre U_2 ne suffira plus pour les orthogonaliser.

Le filtre de prédiction dépend des deux paramètres ρ_0 et ρ_1 . En outre, la minimisation de la norme L_1 revient à résoudre le critère suivant :

$$\min_{\{\rho_0, \rho_1\}} \|C^j\|_1. \quad (3.15)$$

Vu que le premier filtre de mise à jour U_1 est effectué avant l'étape de prédiction, il en résulte donc que les coefficients d'ondelettes dépendent des deux filtres P et U_1 . Ainsi, nous devons optimiser simultanément les filtres de prédiction P et de mise à jour U_1 . L'expression du critère (3.15) peut se réécrire comme suit :

$$\min_{\{\rho_0, \rho_1\}} \|V_1^j - P\left(\frac{V_0^j - U_1(V_1^j)}{\beta}\right)\|_1. \quad (3.16)$$

Pour respecter la contrainte du schéma de subdivision de Loop, nous supposons que la somme des quatre poids du masque de prédiction est égale à 1. En minimisant ce critère, les poids δ de U_1 et β sont recalculés à chaque itération, puisqu'ils dépendent de ρ_0 et ρ_1 [15]. Pour un sommet v appartenant à V_0^j , ils sont donnés par :

$$\beta(n) = \frac{1}{1 - \rho_0} \alpha(n) - \frac{\rho_0}{1 - \rho_0}, \quad (3.17)$$

et

$$\delta(n) = \frac{1 - \beta(n)}{n} \quad (3.18)$$

où n est la valence de v à cette résolution et α est une fonction, qui est définie dans [94] :

$$\alpha(n) = \rho_0 + \left(\rho_0 + 2\rho_1 \cos \frac{2\pi}{n}\right)^2. \quad (3.19)$$

Pour déterminer ce minimum, nous avons utilisé l'algorithme de simplex de Nelder-Mead [85] (voir annexe 6.6). Notons toutefois que d'autres types d'algorithmes peuvent être utilisés, en vue de déterminer ce minimum.

3.6.2 Calcul du deuxième filtre de mise à jour

Une fois les filtres de prédiction P et de mise à jour U_1 sont optimisés, il faut modifier le deuxième filtre de mise à jour U_2 . Une étude détaillée permettant de calculer les

poids ω de U_2 est donnée dans [15], en vue d'obtenir une transformée en ondelettes biorthogonales. Les poids ω et U_2 dépendent des poids ρ du filtre de prédiction. Vu que les valeurs de ces poids sont modifiées, nous devons recalculer des nouveaux poids ω .

Pour ceci, nous utilisons le même raisonnement que celui présenté dans [15], en prenant en compte les nouvelles valeurs ρ^* , qui sont calculées afin de minimiser la norme $L1$ des coefficients, et ce, comme expliqué précédemment. Ainsi, pour obtenir les poids ω du deuxième filtre de mise à jour U_2 , nous devons résoudre l'équation suivante : $A\omega = b$, avec A une matrice symétrique (4×4) , telle que :

$$A = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix},$$

et b un vecteur de taille 4, tel que :

$$b = [b_0 \ b_1 \ b_2 \ b_3], \quad (3.20)$$

où les paramètres a_{ij} et b_i sont définis par les produits scalaires suivants :

$$a_{ij} = \langle \phi_i, \phi_j \rangle, \quad (3.21)$$

et

$$b_i = \langle \psi_l, \phi_i \rangle, \quad (3.22)$$

avec ϕ_i la i ème fonction d'échelle ($i = 0 \dots 4$) de la transformée basée Loop et ψ_l la fonction d'ondelettes. Nous proposons de traiter l'exemple suivant : $b_1 = \langle \psi_l, \phi_1 \rangle$, en vue d'expliquer la procédure de calcul des deux paramètres a_{ij} et b_i . La figure 3.14 montre la fonction d'échelle discrète ϕ_1 et la fonction d'ondelettes avec les paramètres associés. Notons que la i ème fonction d'échelle ϕ_i est associée au poids ω_i (voir figures 3.12(b) et 3.13). En superposant ces deux masques et en calculant le produit scalaire discret, nous pouvons résoudre aisément l'expression suivante : $b_i = \gamma_0 \delta_0 + \alpha_1 \delta_1 + \rho_0$, où :

$$\alpha_i = \alpha(n_i), \quad (3.23)$$

$$\beta_i = \beta(n_i), \quad (3.24)$$

$$\gamma_i = \frac{1 - \alpha_i}{n_i}, \quad (3.25)$$

$$\delta_i = \frac{1 - \beta_i}{n_i}, \quad (3.26)$$

avec n_i la valence du sommet associé.

D'une manière similaire, la résolution de l'équation suivante : $a_{11} = \gamma_0^2 + \alpha_1^2 + \gamma_2^2 + \gamma_3^2 + (n_1 - 3)\gamma_6^2 + n_1(\rho_0^2 + \rho_1^2)$ peut être effectuée en superposant le masque de ϕ_1 sur lui même. Notons toutefois que nous avons opté ici pour la même démarche que celle

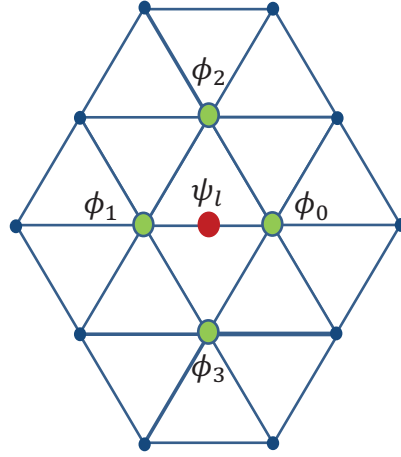


FIGURE 3.13 – Positions des quatre fonctions d'échelle et de la fonction d'ondelettes de la transformée basée Loop.

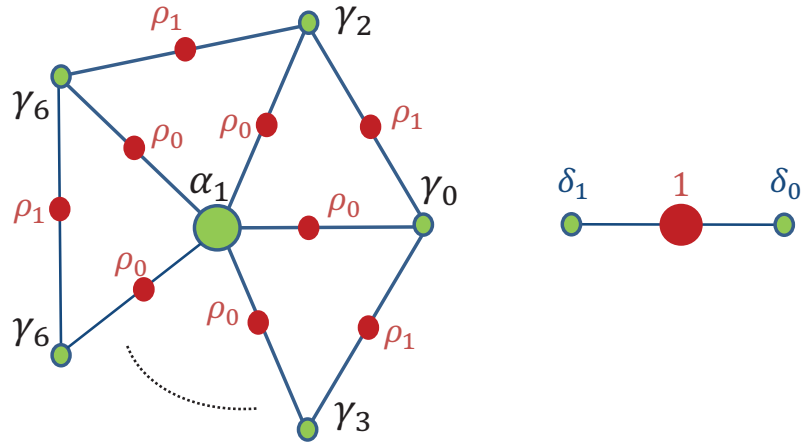


FIGURE 3.14 – Gauche : Fonction d'échelle discrète ϕ_1 de la transformée basée Loop. Droite : Fonction d'ondelettes ψ_l de la transformée basée Loop.

de Bertram [15], en supposant que les valences des sommets, qui ne se trouvent pas à l'intérieur du masque de Loop, sont égales à 6 (i.e., les points bleus de la figure 3.13).

En utilisant le même raisonnement pour tous les paramètres a_{ij} et b_i , nous pouvons calculer les éléments de la matrice A à partir des expressions suivantes :

$$a_{00} = \alpha_0^2 + \gamma_1^2 + \gamma_2^2 + \gamma_3^2 + (n_0 - 3)\gamma_6^2 + n_0(\rho_0^2 + \rho_1^2) \quad (3.27)$$

$$a_{01} = a_{10} = \alpha_0\gamma_0 + \gamma_1\alpha_1 + \gamma_2^2 + \gamma_3^2 + \rho_0^2 + 4\rho_0\rho_1 \quad (3.28)$$

$$a_{02} = a_{20} = \alpha_0\gamma_0 + \gamma_1^2 + \gamma_2\alpha_2 + \gamma_6^2 + \rho_0^2 + 4\rho_0\rho_1 \quad (3.29)$$

$$a_{03} = a_{30} = \alpha_0\gamma_0 + \gamma_1^2 + \gamma_6^2 + \gamma_3\alpha_3 + \rho_0^2 + 4\rho_0\rho_1 \quad (3.30)$$

$$a_{11} = \gamma_0^2 + \alpha_1^2 + \gamma_2^2 + \gamma_3^2 + (n_1 - 3)\gamma_6^2 + n_1(\rho_0^2 + \rho_1^2) \quad (3.31)$$

$$a_{12} = a_{21} = \gamma_0^2 + \alpha_1\gamma_1 + \gamma_2\alpha_2 + \gamma_6^2 + \rho_0^2 + 4\rho_0\rho_1 \quad (3.32)$$

$$a_{13} = a_{31} = \gamma_0^2 + \alpha_1\gamma_1 + \gamma_6^2 + \gamma_3\alpha_3 + \rho_0^2 + 4\rho_0\rho_1 \quad (3.33)$$

$$a_{22} = \gamma_0^2 + \gamma_1^2 + \alpha_2^2 + (n_2 - 2)\gamma_6^2 + n_2(\rho_0^2 + \rho_1^2) \quad (3.34)$$

$$a_{23} = a_{32} = \gamma_0^2 + \gamma_1^2 + \rho_1^2 \quad (3.35)$$

$$a_{33} = \gamma_0^2 + \gamma_1^2 + \alpha_3^2 + (n_3 - 2)\gamma_6^2 + n_3(\rho_0^2 + \rho_1^2). \quad (3.36)$$

De même, nous pouvons calculer les éléments du vecteur b à partir des expressions suivantes :

$$b_0 = (\alpha_0\delta_0 + \gamma_1\delta_1 + \rho_0) \quad (3.37)$$

$$b_1 = (\gamma_0\delta_0 + \alpha_1\delta_1 + \rho_0) \quad (3.38)$$

$$b_2 = b_3 = (\gamma_0\delta_0 + \gamma_1\delta_1 + \rho_1). \quad (3.39)$$

Ainsi, nous obtenons les poids ω , qui sont relatifs à chaque sommet et à chaque résolution, en effectuant l'opération suivante (nous avons utilisé la méthode du pivot de Gauss pour déterminer l'inverse de la matrice A) :

$$\omega = A^{-1}b. \quad (3.40)$$

3.6.3 Validation de la méthode proposée

Cette sous-section est dédiée à la validation de la méthode que nous venons de proposer pour le schéma de Loop. Pour cela, nous comparons les normes $L1$ des coefficients d'ondelettes obtenus, en utilisant notre méthode d'optimisation et le schéma développé par Bertram [15]. Les tableaux 3.5, 3.6, et 3.7 donnent les résultats numériques obtenus pour les objets VASE LION, BIMBA et JOAN OF ARC. Les tableaux 3.5, 3.6 et 3.7 donnent les résultats obtenus. La colonne "Classique" donne les valeurs de la norme $L1$ des composantes tangentielles (C.T.) et de la composante normale (C.N.) des coefficients d'ondelettes à chaque niveau de résolution, en utilisant le schéma basé Loop de Bertram [15]. La colonne "Adaptative" montre les mêmes valeurs lorsque la transformée adaptative proposée est appliquée.

TABLE 3.5 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielles ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet VASE LION, avec ou sans l'algorithme d'optimisation proposé. res est le niveau de résolution.

res	Classique		Adaptative	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	10,08	12,54	5,55	7,93
2	15,70	26,34	10,89	21,39
3	20,37	39,39	16,55	35,02
4	22,93	42,25	19,73	38,44
5	23,39	40,49	21,41	37,96
6	17,71	32,76	17,49	32,71

TABLE 3.6 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielles ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet JOAN OF ARC, avec ou sans l'algorithme d'optimisation proposé. res est le niveau de résolution.

res	Classique		Adaptative	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	13,63	15,51	8,05	11,65
2	16,49	17,24	13,18	14,96
3	16,02	16,85	13,78	15,37
4	15,89	16,91	14,74	16,06
5	12,70	13,82	12,56	13,80

TABLE 3.7 – Comparaison des normes $L1$ des ensembles de coefficients d'ondelettes (composantes tangentielles ($C.T.$) et composante normale ($C.N.$)) à chaque niveau de résolution pour l'objet BIMBA, avec ou sans l'algorithme d'optimisation proposé. res est le niveau de résolution.

res	Classique		Adaptative	
	$C.T.$	$C.N.$	$C.T.$	$C.N.$
1	3,75	6,20	1,98	4,32
2	5,51	10,25	3,92	8,34
3	6,29	14,29	4,98	12,66
4	5,94	15,07	5,18	14,28
5	4,51	11,20	4,20	10,67
6	5,71	16,26	5,27	15,06
6	4,79	15,20	4,73	15,16

D'après ces tableaux nous constatons que l'utilisation de la méthode d'optimisation proposée basé sur un schéma adaptatif permet d'avoir une valeur de la norme $L1$ des coefficients d'ondelettes globalement plus petite que celle obtenue par celle développée par Bertram [15]. Cette minimisation de la norme $L1$ des coefficients obtenus, permettra d'améliorer les taux de compression, comme nous allons le voir dans la section suivante.

3.7 Application à la compression

Pour montrer l'intérêt de notre méthode d'optimisation des schémas basés Butterfly et Loop, nous avons intégré la transformée adaptative proposée dans deux codeurs géométriques de l'état de l'art [79, 112]. Ainsi, nous avons évalué les gains apportés par le schéma proposé en terme de compression.

3.7.1 Principe d'un codeur géométrique

Le principe d'un codeur géométrique est donné figure 3.15.

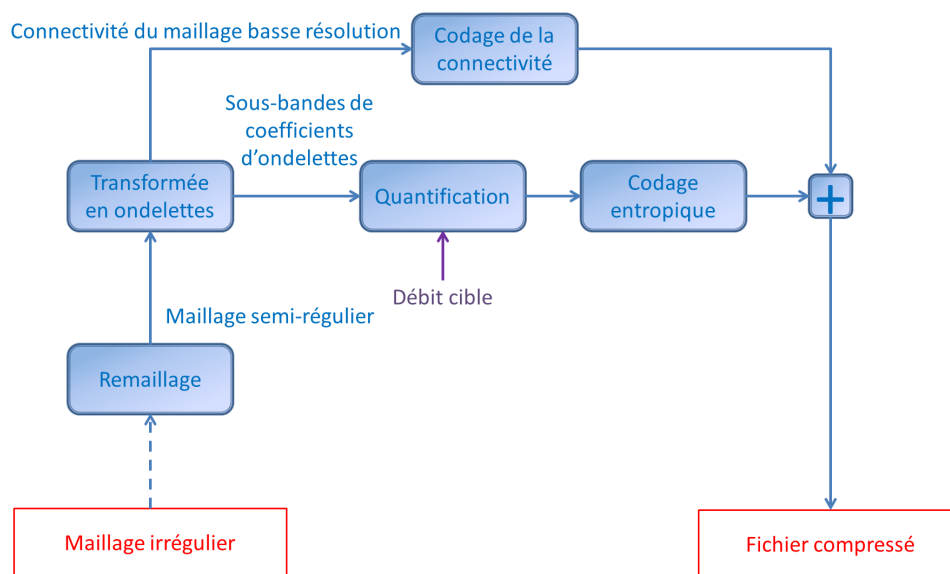


FIGURE 3.15 – Principe général d'un codeur géométrique.

Les étapes principales d'un codeur géométrique sont :

1. remaillage, où le maillage irrégulier original est remaillé pour obtenir un maillage semi-régulier ;
2. transformée en ondelettes, où une transformée en ondelettes à N niveaux est appliquée sur le maillage semi-régulier pour obtenir les N sous-bandes de coefficients d'ondelettes et le maillage BF ;

3. codage de la connectivité, où la connectivité du maillage BF est codée sans pertes (par exemple, avec le codeur proposé dans [133]) ;
4. quantification, dans laquelle les N sous-bandes de coefficients d'ondelettes sont quantifiées afin d'obtenir l'erreur de reconstruction la plus petite pour un débit donné par l'utilisateur (ou inversement, afin d'obtenir le débit le plus petit pour une erreur de reconstruction donnée) ;
5. codage entropique, où les coefficients quantifiés sont codés avec un codeur entropique. La trame binaire finale sera composée par ces coefficients quantifiés et codés, la connectivité codée et des informations supplémentaires (header), qui seront nécessaires au décodage de cette trame.

3.7.2 Description des codeurs testés

Nous avons utilisé deux codeurs lors de nos tests, à savoir :

- le codeur PGC, qui est développé dans [79] ;
- le codeur EDQ, qui est développé dans [112, 113].

Codeur PGC

Le codeur PGC [79] est le premier codeur basé ondelettes pour les maillages semi-réguliers (2000). Ce codeur, qui est actuellement le plus utilisé, est doté des deux principales caractéristiques suivantes :

- un codage par zero-tree efficace des coefficients d'ondelettes,
- un décodage et une transmission progressifs.

Pour plus de détails sur le codeur PGC, voir [79].

Codeur EDQ

Le codeur EDQ a été proposé en 2005. Ses principales caractéristiques sont :

- un processus d'allocation débit qui optimise la quantification. Le principe de ce processus est de calculer les pas de quantifications optimaux $\{q_i\}$ de chaque sous-bande pour un débit donné R_{cible} , en résolvant le problème suivant :

$$(\mathcal{P}) \begin{cases} \text{minimiser} & D_T(\{q_i\}) \\ \text{avec} & R_T(\{q_i\}) = R_{cible} \end{cases} \quad (3.41)$$

$$(3.42)$$

avec D_T la EQM (Erreur Quadratique Moyenne) entre les sommets originaux et les sommets quantifiés et R_T le débit total obtenu ;

- ce problème sous contraintes peut être défini avec un critère de Lagrangien, tel que :

$$J_\lambda(\{q_i\}) = D_T(\{q_i\}) + \lambda(R_T(\{q_i\}) - R_{vible}), \quad (3.43)$$

- ce processus d'allocation débit est basé modèles. Des modèles théoriques sont utilisés pour estimer le débit et la distorsion de chaque sous-bande évitant, ainsi, le codage réel des données à de nombreuses reprises juste pour calculer le débit et la distorsion de chaque sous-bande à chaque itération du processus ;
 - ces modèles utilisés supposent que les densités de probabilité des coefficients d'ondelettes peuvent être modélisées par des Gaussiennes généralisées.
- Pour plus de détails sur le codeur EDQ, voir [112, 113].

3.7.3 Résultats pour le schéma basé Butterfly

Dans cette sous-section, nous présentons les résultats obtenus pour le schéma basé Butterfly. Ces résultats sont répartis en deux catégories. Il s'agit des résultats numériques et des résultats visuels.

Résultats numériques

Les figures 3.16 et 3.17 montrent les courbes débit/PSNR (relatif à la distance surface-surface d_S présentée dans la section 1.5 du chapitre 1) pour VASE LION quand nous utilisons le schéma classique et le schéma adaptatif proposés, et ce, avec les codeurs EDQ et PGC, respectivement. Le PSNR est donné par :

$$PSNR(dB) = 20 \log_{10} \frac{BB}{d_S}, \quad (3.44)$$

avec BB la diagonale de la boîte englobante. Les débits sont donnés en bits par sommet irrégulier (b/iv). Pour l'approche adaptative, les débits incluent aussi l'ensemble des poids des filtres utilisés (ces informations sont nécessaires lors du décodage). Les tableaux 3.8, 3.9 et 3.10 donnent les valeurs numériques (PSNR et d_S) pour ce même modèle, mais aussi pour BIMBA et RABBIT, en utilisant le codeur PGC. Les tableaux 3.11, 3.12 et 3.13 donnent les mêmes valeurs en utilisant le codeur EDQ.

Nous remarquons que l'utilisation de notre méthode d'optimisation, qui est basée sur un schéma adaptatif, permet d'obtenir globalement un gain positif en terme de d_S . Les gains moyens obtenus pour les deux codeurs (PGC et EDQ) sont : $\sim 4,12\%$ pour VASE LION, $\sim 2,87\%$ pour BIMBA et $\sim 0,94\%$ pour RABBIT. Les gain maximaux sont : $\sim 6,29\%$ pour VASE LION, $\sim 7,33\%$ pour BIMBA et $\sim 2,01\%$ pour RABBIT. Nous remarquons que les gains sont similaires pour les deux codeurs. Nous remarquons aussi que les gains les plus significatifs sont obtenus avec VASE LION et BIMBA, en particulier à bas et moyen débits. Ceci s'explique par le fait qu'il y a le plus de marge de progression dans ces gammes de débits. Le gain apporté par notre méthode est donc plus significatif dans ces débits. Nous constatons que les gains obtenus avec RABBIT sont assez faibles. Ceci s'explique par le fait que RABBIT est un objet lisse, et que notre méthode a été justement développée pour les surfaces détaillées et surtout non lisses. Pour un modèle comme RABBIT, la transformée en ondelettes classiques est suffisante et notre méthode améliore très légèrement le

TABLE 3.8 – Résultats de compression obtenus pour VASE LION en utilisant le schéma basé Butterfly et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
2,5	$5,49 \times 10^{-4}$	68,03
3,5	$4,08 \times 10^{-4}$	70,60
5,5	$2,70 \times 10^{-4}$	74,20
9	$1,84 \times 10^{-4}$	77,52
11	$1,58 \times 10^{-4}$	78,86
13	$1,46 \times 10^{-4}$	79,54

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
2,5	$5,19 \times 10^{-4}$	68,51	5,37%	$5,27 \times 10^{-4}$	68,38	3,95%
3,5	$3,82 \times 10^{-4}$	71,16	6,29%	$3,87 \times 10^{-4}$	71,07	5,27%
5,5	$2,57 \times 10^{-4}$	74,62	4,75%	$2,58 \times 10^{-4}$	74,57	4,12%
9	$1,78 \times 10^{-4}$	77,79	3%	$1,79 \times 10^{-4}$	77,75	2,59%
11	$1,53 \times 10^{-4}$	79,14	3,13%	$1,54 \times 10^{-4}$	79,07	2,44%
13	$1,40 \times 10^{-4}$	79,91	4,22%	$1,40 \times 10^{-4}$	79,88	3,91%

TABLE 3.9 – Résultats de compression obtenus pour BIMBA en utilisant le schéma basé Butterfly et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
1,75	$1,21 \times 10^{-4}$	81,91
2,5	$8,43 \times 10^{-5}$	85,04
3,5	$6,36 \times 10^{-5}$	87,79
5,5	$4,76 \times 10^{-5}$	90,01
9	$3,76 \times 10^{-5}$	92,06
11	$3,46 \times 10^{-5}$	92,78

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
1,75	$1,14 \times 10^{-4}$	82,38	5,28%	$1,15 \times 10^{-4}$	82,32	4,59%
2,5	$8,24 \times 10^{-5}$	85,24	2,2%	$8,31 \times 10^{-5}$	85,17	1,41%
3,5	$6,31 \times 10^{-5}$	87,56	0,77%	$6,32 \times 10^{-5}$	87,54	0,54%
5,5	$4,65 \times 10^{-5}$	90,21	2,31%	$4,67 \times 10^{-5}$	90,18	1,96%
9	$3,70 \times 10^{-5}$	92,18	1,42%	$3,71 \times 10^{-5}$	92,17	1,22%
11	$3,43 \times 10^{-5}$	92,85	0,87%	$3,44 \times 10^{-5}$	92,83	0,58%

TABLE 3.10 – Résultats de compression obtenus pour RABBIT en utilisant le schéma basé Butterfly et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
1,75	$7,81 \times 10^{-6}$	81,53
2,5	$6,41 \times 10^{-6}$	83,25
3,5	$5,69 \times 10^{-6}$	84,29
5,5	$5,04 \times 10^{-6}$	85,33
9	$4,61 \times 10^{-6}$	86,11
13	$4,48 \times 10^{-6}$	86,35

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
1,75	$7,66 \times 10^{-6}$	81,70	2,01%	$7,67 \times 10^{-6}$	81,68	1,79%
2,5	$6,38 \times 10^{-6}$	83,29	0,43%	$6,40 \times 10^{-6}$	83,26	0,12%
3,5	$5,59 \times 10^{-6}$	84,43	1,65%	$5,62 \times 10^{-6}$	84,40	1,25%
5,5	$4,99 \times 10^{-6}$	85,43	1,06%	$5,01 \times 10^{-6}$	85,40	0,74%
9	$4,59 \times 10^{-6}$	86,15	0,46%	$4,59 \times 10^{-6}$	86,15	0,43%
13	$4,48 \times 10^{-6}$	86,36	0,14%	$4,48 \times 10^{-6}$	86,37	0,15%

TABLE 3.11 – Résultats de compression obtenus pour VASE LION en utilisant le schéma basé Butterfly et le codeur EDQ. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
2,58	$5,06 \times 10^{-4}$	68,73
3,51	$3,87 \times 10^{-4}$	71,07
5,33	$2,78 \times 10^{-4}$	73,92
8,90	$1,78 \times 10^{-4}$	77,79
10,78	$1,62 \times 10^{-4}$	78,63
12,70	$1,45 \times 10^{-4}$	79,56

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
2,58	$4,83 \times 10^{-4}$	69,14	4,68%	$4,82 \times 10^{-4}$	69,15	4,78%
3,51	$3,78 \times 10^{-4}$	71,27	2,33%	$3,73 \times 10^{-4}$	71,39	3,59%
5,33	$2,71 \times 10^{-4}$	74,17	2,77%	$2,70 \times 10^{-4}$	74,20	3,14%
8,90	$1,71 \times 10^{-4}$	78,15	4,01%	$1,71 \times 10^{-4}$	78,17	4,22%
10,78	$1,53 \times 10^{-4}$	79,11	5,34%	$1,53 \times 10^{-4}$	79,13	5,61%
12,70	$1,40 \times 10^{-4}$	79,88	3,60%	$1,40 \times 10^{-4}$	79,89	3,71%

TABLE 3.12 – Résultats de compression obtenus pour Bimba en utilisant le schéma basé Butterfly et le codeur EDQ. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
1,55	$1,39 \times 10^{-4}$	80,70
2,17	$9,86 \times 10^{-5}$	83,68
3,03	$7,03 \times 10^{-5}$	86,62
4,75	$5,15 \times 10^{-5}$	89,32
7,85	$3,94 \times 10^{-5}$	91,64
9,60	$3,62 \times 10^{-5}$	92,38

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
1,55	$1,29 \times 10^{-4}$	81,36	7,33%	$1,29 \times 10^{-4}$	81,33	7,02%
2,17	$9,21 \times 10^{-5}$	84,27	6,58%	$9,31 \times 10^{-5}$	84,18	5,51%
3,03	$6,87 \times 10^{-5}$	86,82	2,23%	$6,80 \times 10^{-5}$	86,91	3,25%
4,75	$5,02 \times 10^{-5}$	89,54	2,42%	$5,00 \times 10^{-5}$	89,58	2,86%
7,85	$3,90 \times 10^{-5}$	91,74	1,09%	$3,90 \times 10^{-5}$	91,74	1,13%
9,60	$3,55 \times 10^{-5}$	92,54	1,90%	$3,56 \times 10^{-5}$	92,54	1,80%

TABLE 3.13 – Résultats de compression obtenus pour RabbIt en utilisant le schéma basé Butterfly et le codeur EDQ. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique	
	d_S	PSNR (dB)
1,07	$9,96 \times 10^{-6}$	79,42
1,89	$7,12 \times 10^{-6}$	82,34
2,63	$6,45 \times 10^{-6}$	83,19
3,65	$5,46 \times 10^{-6}$	84,64
10,20	$4,51 \times 10^{-6}$	86,30

Débit	Méthode OaC			Méthode OsC		
	d_S	PSNR (dB)	Gain	d_S	PSNR (dB)	Gain
1,07	$9,79 \times 10^{-6}$	79,573	1,72%	$9,80 \times 10^{-6}$	79,56	1,55%
1,89	$7,03 \times 10^{-6}$	82,450	1,23%	$7,07 \times 10^{-6}$	82,40	0,62%
2,63	$6,39 \times 10^{-6}$	83,277	1,00%	$6,38 \times 10^{-6}$	83,28	1,07%
3,65	$5,44 \times 10^{-6}$	84,670	0,40%	$5,42 \times 10^{-6}$	84,70	0,79%
10,20	$4,50 \times 10^{-6}$	86,25	0,29%	$4,50 \times 10^{-6}$	86,32	0,18%

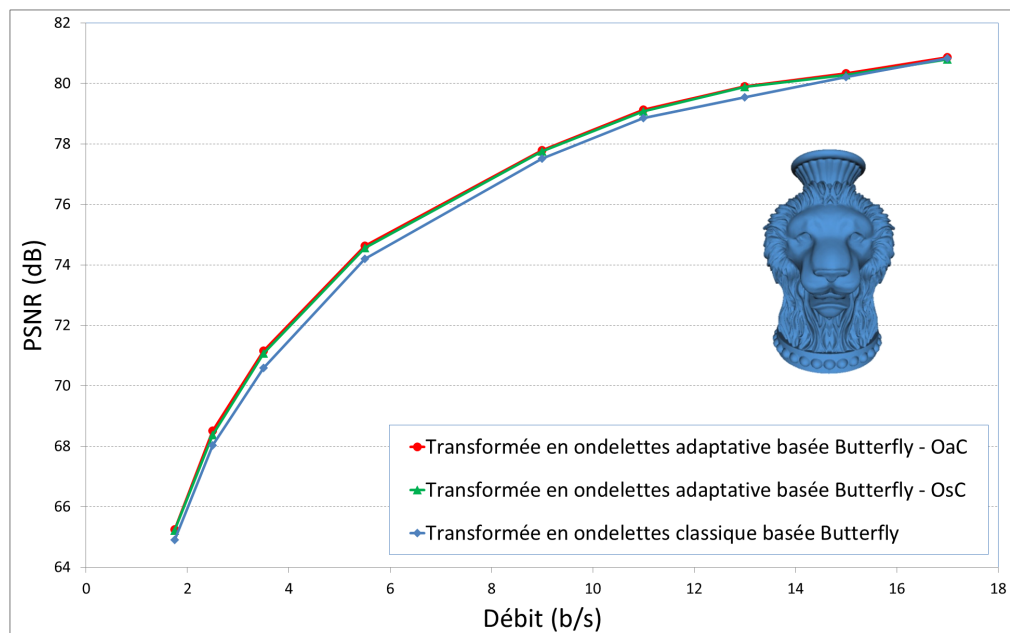


FIGURE 3.16 – Courbes d'évolution du PSNR pour VASE LION lorsque le schéma basé Butterfly et le codeur PGC sont utilisés.

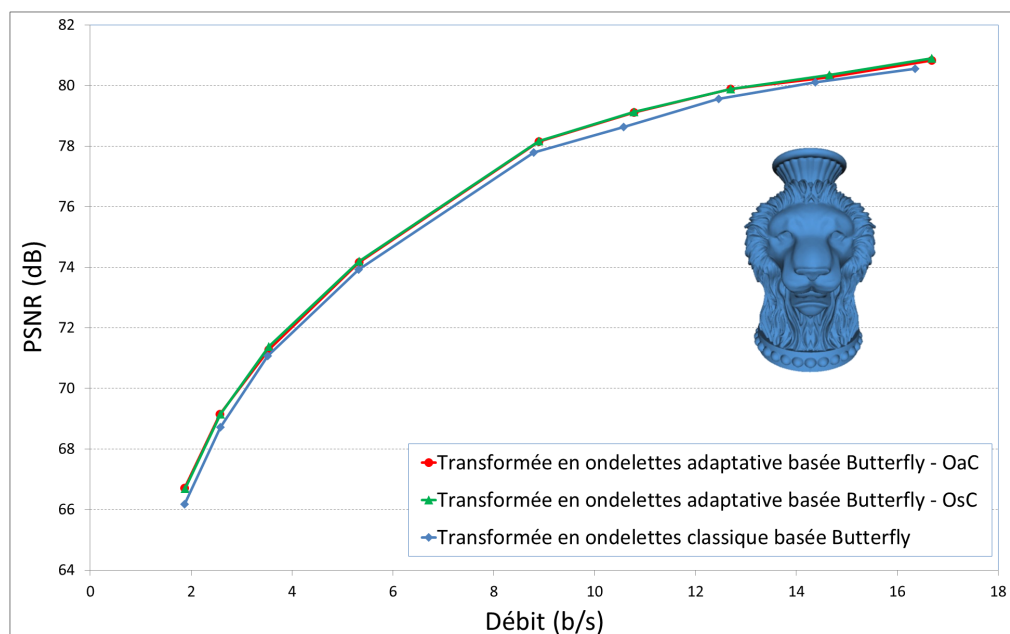


FIGURE 3.17 – Courbes d'évolution de PSNR pour VASE LION lorsque le schéma basé Butterfly et le codeur EDQ sont utilisés.

codage de ce modèle. Ces résultats de simulation numérique confirment bien les résultats théoriques développés. Soulignons que, plus la surface est détaillée et non lisse, plus notre approche est pertinente et donc performante.

Remarquons que notre méthode donne des résultats similaires en utilisant l'approche avec ou sans contraintes sur la symétrie des filtres utilisés. Sachant que l'optimisation avec contraintes est trois fois plus rapide que celle sans contraintes (voir tableau 3.14), nous avons finalement opté pour cette variante.

TABLE 3.14 – Temps de calcul pour l'optimisation du schéma basé Butterfly (implémentation avec MATLAB).

modèle	# résolution	# sommets	OaC	OsC
BIMBA	7	999426	79 sec	207 sec
RABBIT	6	163842	13 sec	34 sec
VASE LION	6	675842	55 sec	141 sec

Résultats visuels

La figure 3.7.4 montre quelques améliorations visuelles apportées par notre approche adaptative avec (figure 3.18(b)) ou sans (figure 3.18(a)) l'utilisation de notre schéma adaptatif. La couleur sur ces figures correspond aux distances "point-surface", normalisées par la diagonale de la boîte englobante, entre le maillage originale et le maillage codé. Remarquons que notre approche adaptative préserve mieux les régions détaillées. Par contre, sur les zones lisses l'approche classique donne une plus faible erreur. Ces résultats visuels confirment que notre approche est bien adaptée pour les régions détaillées et non lisses. Nous pouvons donc conclure qu'il serait intéressant de combiner les deux transformées (classique et adaptative) dans un schéma de codage basé régions. Par exemple, en utilisant une technique de segmentation, nous pourrions diviser le maillage en deux parties (lisse et rugueuse), qui seront analysées séparément (région lisse = transformée classique, région rugueuse = transformée adaptative). Les gains en compression et en qualité visuelle devraient être conséquents. Ceci fait partie des perspectives de ces travaux.

Nous avons aussi comparé l'efficacité de notre schéma adaptatif et du schéma classique sur des maillages avec arêtes saillantes. La figure 3.7.4 montre que notre schéma adaptatif préserve aussi mieux les arêtes saillantes que le schéma classique pour le modèle KNOT.

3.7.4 Résultats pour le schéma basé Loop

Cette sous-section est relative à la présentation des résultats obtenus pour le schéma basé Loop. Nous avons réparti ces résultats en deux catégories. Il s'agit des résultats numériques et des résultats visuels.

Résultats numériques

La figure 3.20 montre les courbes d'évolution du débit/PSNR (relatif à la distance surface-surface d_S présentée dans la section 1.5 du chapitre 1) pour VASE LION quand le schéma classique et notre schéma optimisé avec le codeur PGC sont utilisés. Les tableaux 3.15, 3.16 et 3.17 donnent les valeurs numériques (PSNR et d_S) pour ce même modèle, mais aussi pour BIMBA et RABBIT en utilisant le codeur PGC.

Remarquons que l'utilisation de notre schéma adaptatif permet d'obtenir globalement un gain positif en terme de distance surface-surface. Les gains moyens obtenus sont : $\sim 8,09\%$ pour VASE LION, $\sim 8,73\%$ pour BIMBA et $\sim 4,63\%$ pour RABBIT. Les gains maximaux sont : $\sim 19,55\%$ pour VASE LION, $\sim 19,73\%$ pour BIMBA et $\sim 6,31\%$ pour RABBIT. Ces résultats montrent que notre approche adaptative donne de meilleurs résultats que le schéma classique.

TABLE 3.15 – Résultats de compression obtenus pour VASE LION en utilisant le schéma basé Loop et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique		Adaptative		
	d_S	PSNR (dB)	d_S	PSNR (dB)	Gain
2,5	$1,53 \times 10^{-3}$	59,13	$1,39 \times 10^{-3}$	59,95	9,01%
5,5	$6,99 \times 10^{-4}$	65,92	$5,63 \times 10^{-4}$	67,81	19,55%
9	$4,47 \times 10^{-4}$	69,80	$4,43 \times 10^{-4}$	69,89	1,08%
11	$3,35 \times 10^{-4}$	72,30	$3,26 \times 10^{-4}$	72,54	2,70%

TABLE 3.16 – Résultats de compression obtenus pour BIMBA en utilisant le schéma basé Loop et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

Débit	Classique		Adaptative		
	d_S	PSNR (dB)	d_S	PSNR (dB)	Gain
1	$4,96 \times 10^{-4}$	69,64	$3,98 \times 10^{-4}$	71,55	19,73%
1,75	$2,71 \times 10^{-4}$	74,90	$2,39 \times 10^{-4}$	76,00	11,86%
3,5	$1,44 \times 10^{-4}$	80,36	$1,41 \times 10^{-4}$	80,55	2,17%
9	$7,76 \times 10^{-5}$	85,76	$7,67 \times 10^{-5}$	85,86	1,16%

Résultats visuels

Il est bien connu que la transformée en ondelettes basée Loop développée par Bertram [15] tend à déformer les maillages obtenus à basses résolutions. C'est d'ailleurs l'un des inconvénients de cette transformée (qui n'est donc pas très adaptée pour être utilisée dans des codeurs progressifs). En comparant les différentes résolutions obtenues, nous remarquons que notre transformée adaptative produit des maillages moins distordus, surtout à basses résolutions, et en particulier avec des objets complexes. Les

TABLE 3.17 – Résultats de compression obtenus pour RABBIT en utilisant le schéma basé Loop et le codeur PGC. Le gain est relatif à la distance surface-surface d_S (voir chapitre 1, section 1.5).

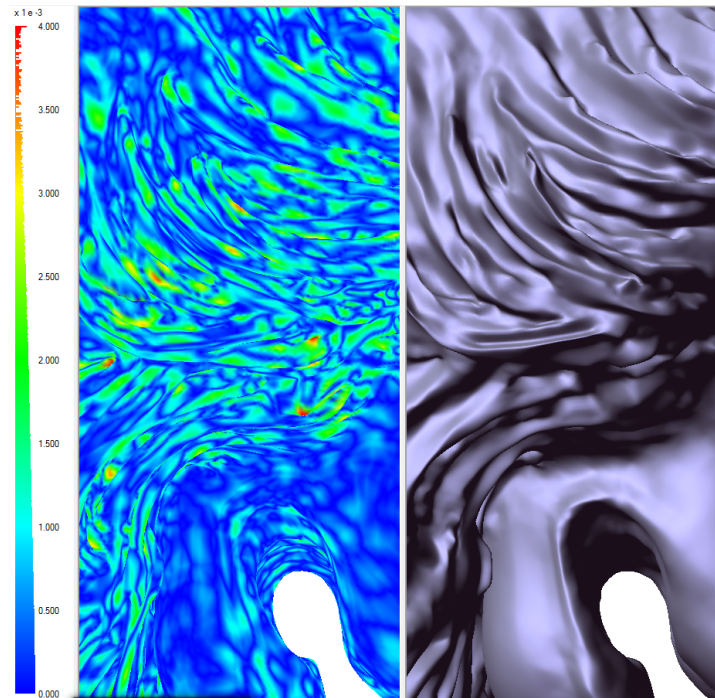
Débit	Classique		Adaptative		
	d_S	PSNR (dB)	d_S	PSNR (dB)	Gain
1,75	$1,01 \times 10^{-5}$	79,34	$9,54 \times 10^{-6}$	79,80	5,18%
5,5	$5,50 \times 10^{-6}$	84,58	$5,47 \times 10^{-6}$	84,63	0,63%
9	$4,79 \times 10^{-6}$	85,79	$4,79 \times 10^{-6}$	85,79	0,38%
11	$4,60 \times 10^{-6}$	86,13	$4,59 \times 10^{-6}$	86,16	0,33%

figures 3.21 et 3.22 montre les différentes résolutions obtenues pour JOAN D'ARC. Nous constatons bien qu'avec notre approche, nous pouvons préserver mieux l'allure générale de l'objet, notamment à basses résolutions. Ceci peut être exploité dans le futur pour des applications progressives (transmission, codage, etc.). Par exemple, dans des applications de transmission progressive, il est préférable d'avoir une bonne approximation de l'objet à basse résolution. Avec notre schéma adaptatif, nous améliorons la qualité de la basse résolution et, par conséquent, nous pouvons donner une meilleure approximation de l'objet.

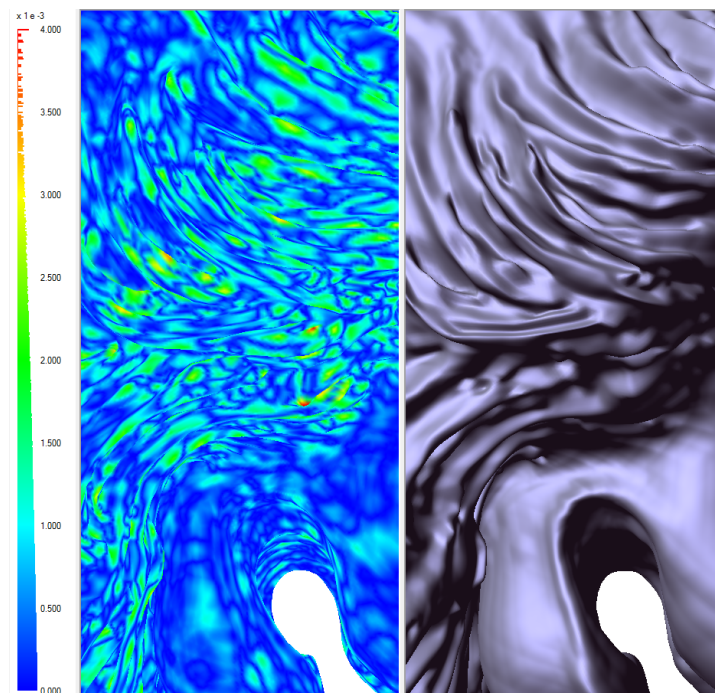
3.8 Conclusion

Nous avons proposé, dans ce troisième chapitre, une méthode d'optimisation de deux transformées en ondelettes basées sur le schéma *lifting* pour les maillages semi-réguliers, à savoir : les transformées basées Butterfly et Loop. Cette méthode consiste à calculer des filtres de prédiction et de mise à jour adaptés au maillage en entrée. Un schéma adaptatif a été proposé, en vue de maximiser la parcimonie des coefficients d'ondelettes pour améliorer les taux de compression. Notons que l'optimisation se fait à chaque niveau de décomposition lors de l'analyse (obtention des filtres P et U optimisés pour chaque niveau de résolution).

Les résultats expérimentaux obtenus ont bien montré que la transformée en ondelettes adaptative proposée améliore les gains de compression pour les modèles globalement non lisses. Nous avons également montré que l'utilisation de notre transformée adaptative basée Butterfly préserve mieux les régions détaillées et les arêtes saillantes. Notre transformée adaptative basée Loop donne des résultats similaires en terme de qualité de reconstruction. Cependant, elle produit des maillages à basses résolutions moins déformés par rapport à la transformée classique, ce qui est très utile dans diverses applications (par exemple, pour une transmission progressive).



(a) Transformée classique.



(b) Transformée adaptative.

FIGURE 3.18 – Distribution de l'erreur géométrique pour VASE LION codé avec PGC et en utilisant les transformées basé Butterfly classique et adaptative.

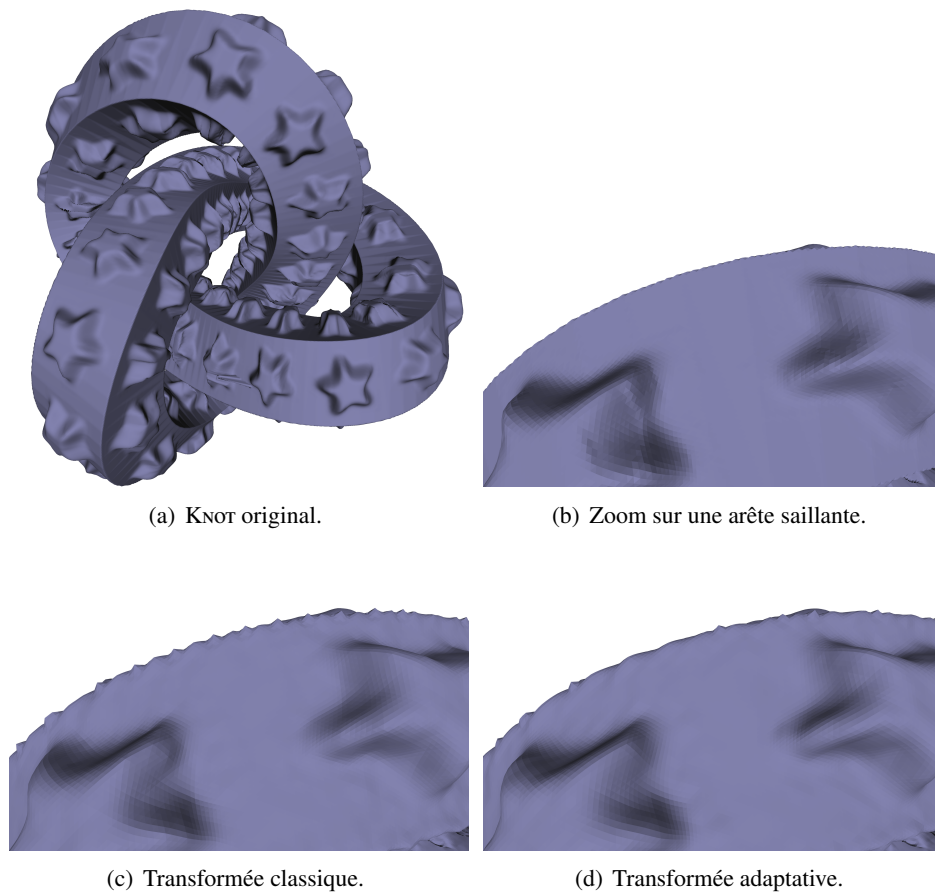


FIGURE 3.19 – Comparaison visuelle pour le modèle KNOT codé avec PGC et en utilisant les transformées basé Butterfly classique et adaptative.

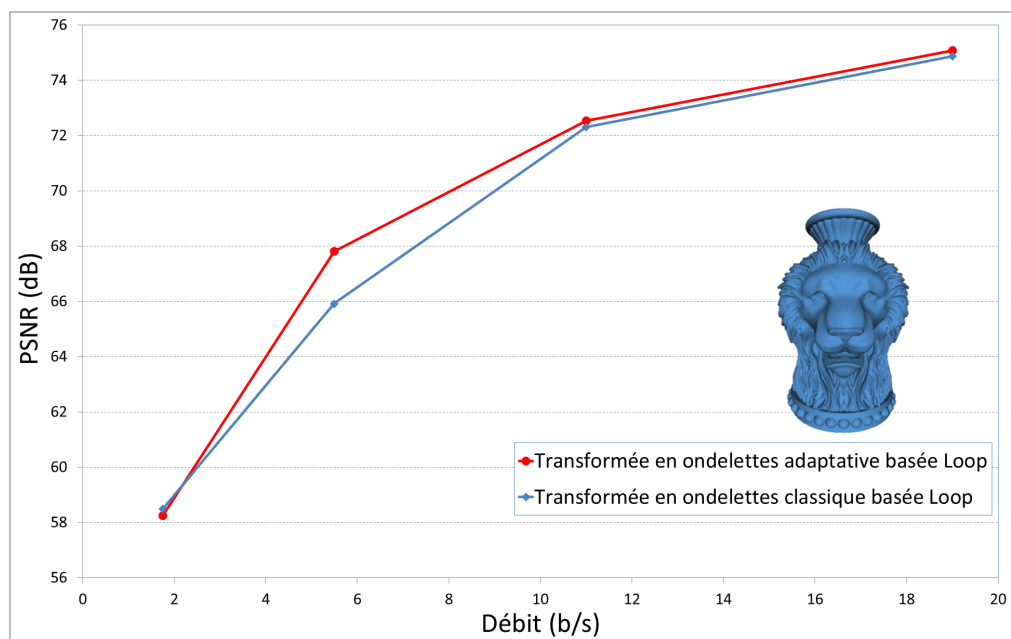


FIGURE 3.20 – Courbes PSNR pour VASE LION lorsqu'on utilise la transformée basée Loop et le codeur PGC.

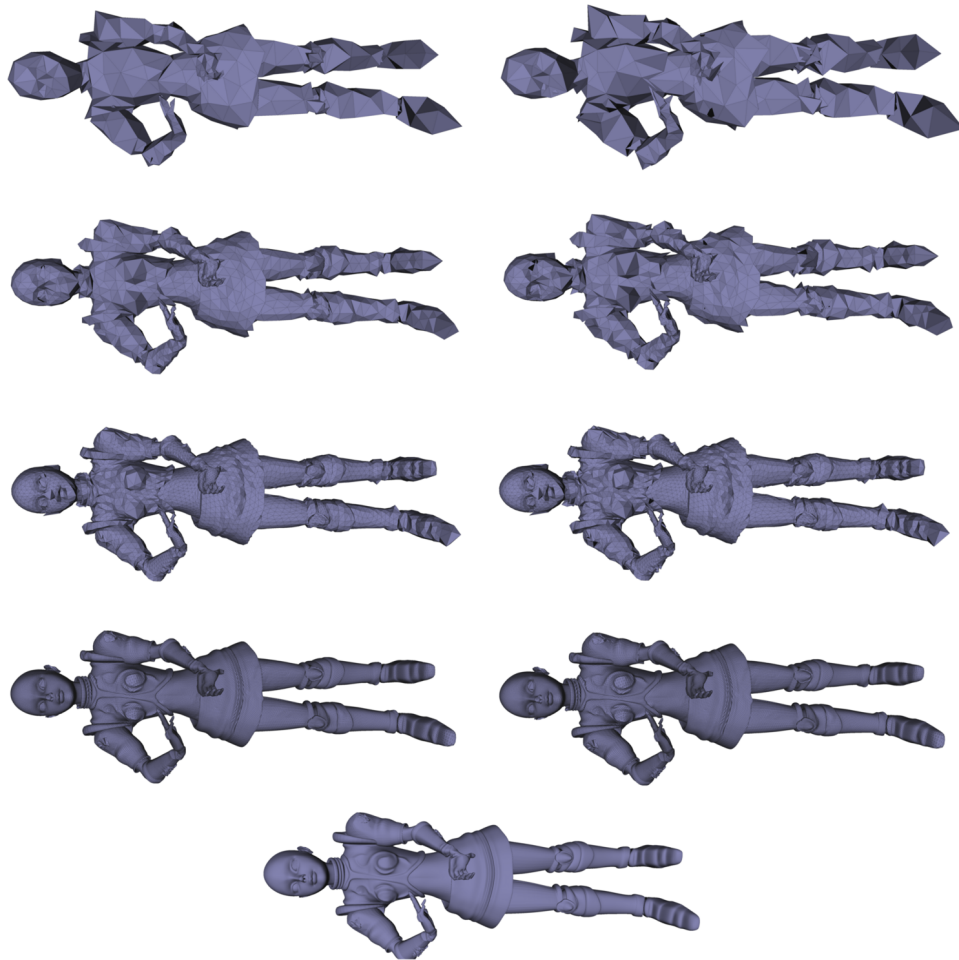


FIGURE 3.21 – Différentes résolutions relatives à la transformée en ondelettes basée Loop. A basse résolution, l'allure du maillage original est généralement mieux préservée avec notre transformée adaptative. Bas : modèle original ; Droite : notre transformée adaptative ; Gauche : transformée classique de [15].

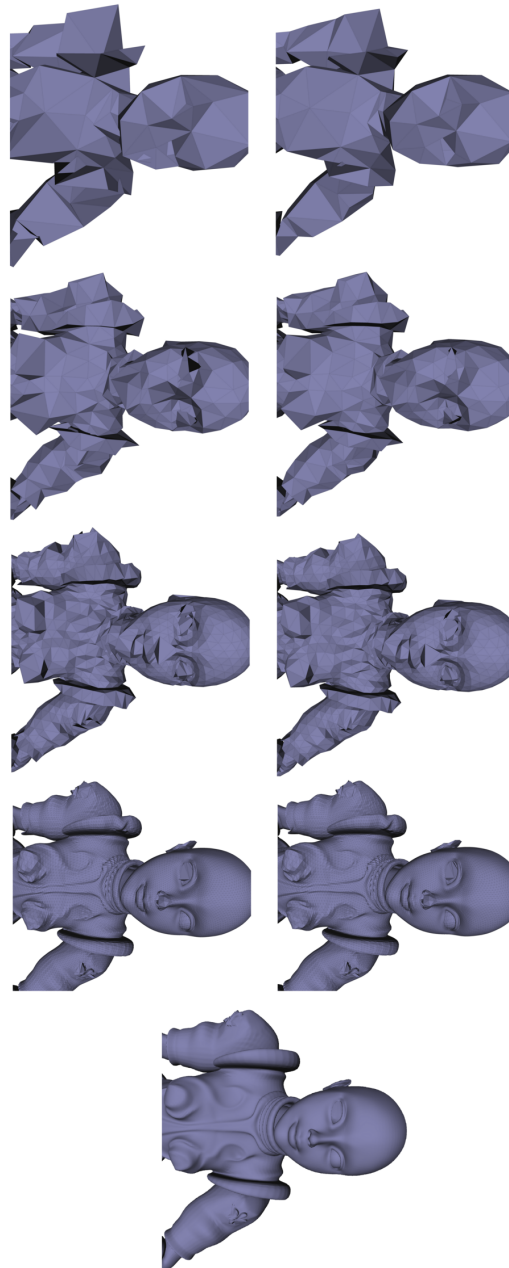


FIGURE 3.22 – Différentes résolutions relatives à la transformée en ondelettes basée Loop. A basse résolution, l'allure du maillage original est généralement mieux préservée avec notre transformée adaptative. Bas : modèle original ; Droite : notre transformée adaptative ; Gauche : transformée classique de [15].

Conclusion

Dans cette première partie du mémoire de notre thèse de doctorat, nous avons développé des méthodes concernant le remaillage semi-régulier et l'optimisation de deux transformées en ondelettes pour les maillages semi-réguliers.

Remaillage semi-régulier

Notre méthode pour le remaillage semi-régulier est basée sur la construction d'un diagramme de Voronoï multirésolution sur le maillage original, et ce, en commençant par construire un diagramme grossier et en lui ajoutant, par la suite, des graines de Voronoï d'une manière adaptative. L'avantage de notre méthode de remaillage réside dans sa simplicité, tout en permettant d'obtenir des maillages adaptatifs de bonne qualité.

Nous envisagerons de traiter dans le futur les trois thèmes suivants :

- gérer les bords et les arêtes saillantes afin de pouvoir remailler les objets avec bords ou les objets CAO ;
- l'application de l'algorithme de remaillage, que nous avons proposé, sur des applications de maillages plus denses et plus complexes ;
- l'adaptation de notre méthode de remaillage semi-régulier pour des nuages de points, en vue de produire des maillages semi-réguliers directement à partir de la sortie de scanners 3D.

Optimisation des transformées en ondelettes

Notre méthode d'optimisation de deux transformées en ondelettes (Butterfly et Loop) basées sur le schéma *lifting* pour les maillages semi-réguliers consiste à maximiser la parcimonie des coefficients d'ondelettes. Pour ceci, nous optimisons les filtres de prédiction à chaque niveau de résolution afin de minimiser la norme $L1$ des coefficients d'ondelettes. Par la suite, nous adaptons les filtres de mise à jour afin de préserver la moyenne du signal (pour la transformée basée Butterfly) ou d'obtenir une transformée biorthogonale (pour la transformée basée Loop).

Nous envisageons de modifier le codeur géométrique proposé par Payan et Antonini [112] afin de le rendre adaptatif et ainsi pouvoir tirer profit de notre méthode de remaillage semi-régulier. En effet, ce codeur exige actuellement qu'en entrée, le maillage semi-régulier soit uniforme. Nous envisageons également d'étudier dans le futur, et dans un premier temps, les problèmes qui sont liés au développement de nouveaux filtres de mise à jour pour la transformée en ondelettes basée Butterfly. Cette étude va permettre de préserver des moments d'ordre supérieur et de définir de nouveaux critères de minimisation. Dans un deuxième temps, nous envisageons de développer une méthode d'analyse basée régions pour les modèles complexes. Comme il a été mentionné un peu plus haut, il pourrait être intéressant d'adapter les filtres pour les différentes régions du maillage d'entrée, et ce, en fonction de la quantité de détails haute fréquence (transformée classique/adaptative pour les régions lisses/non lisses). Une telle méthode devrait permettre d'améliorer la qualité globale des maillages codés avec des codeurs géométriques (en particulier la qualité visuelle). Nous envisageons également de modifier le codeur géométrique EDQ [112, 113] afin qu'il puisse traiter des maillages semi-réguliers adaptatifs.

Deuxième partie

Maillages dynamiques

Introduction

Les objets et scènes 3D sont largement utilisés dans les jeux vidéos, les sites Web, les films, etc. Dans la plupart des cas, un tel contenu est purement virtuel, créé par des opérateurs qualifiés qui conçoivent les décors en utilisant du matériel et des plates-formes logicielles spécifiques. Cela peut impliquer des centaines de personnes qui travaillent pendant des mois pour produire un film avec une excellente qualité. D'autre part, de plus en plus de contenus 3D dynamiques sont utilisés grâce à des nouvelles applications, comme la vidéo avec un point de vue libre. En effet, une scène dynamique du monde réel est généralement capturée par plusieurs caméras synchronisées. La géométrie 3D est reconstruite et représentée dans un format adapté. Puis, un tel contenu peut être utilisé dans n'importe quelle application, comme c'est le cas pour les scènes 3D virtuelles créées par ordinateur.

La géométrie 3D des scènes dynamiques peut être représentée en utilisant des maillages surfaciques. L'approche la plus simple est d'utiliser des maillages complètement indépendants pour chaque instant du temps, c'est-à-dire une succession de maillages statiques. Mais, évidemment ce n'est pas l'approche la plus efficace. En effet, nous pouvons représenter la géométrie des objets 3D animés par une partie constante et une partie dynamique. La partie constante de la représentation reflète le fait qu'un certain objet (virtuel ou reconstitué) ayant des propriétés constantes est affiché. La partie dynamique reflète les propriétés changeantes de cet objet.

Très souvent, les modèles dynamiques 3D sont représentés et produits de cette façon : un maillage initial est créé et, par la suite, déplacé et déformé (en gardant la connectivité constante et en changeant uniquement les positions des sommets) au fil du temps en définissant des paramètres d'animation à partir d'un modèle physique qui décrit l'objet. Une telle représentation est extrêmement compacte. Elle nécessite uniquement un maillage surfacique et une séquence de paramètres d'animation au lieu d'une séquence de maillages surfaciques. Cependant, l'inconvénient est qu'elle nécessite un modèle pré-défini, et est donc limité à un certain type d'objets et/ou déformations.

Dans le cas le plus générique, il n'existe pas de modèle physique. Etant donné un maillage initial, chacun de ses sommets peut être autorisé à se déplacer librement dans le temps, tout en gardant une cohérence temporelle. Une scène 3D dynamique peut être

composée à l'aide de tels maillages animés. Des objets avec un modèle physique et des paramètres d'animation peuvent aussi être intégrés dans une telle scène.

En principe, cette représentation est adaptée et efficace pour n'importe quelle scène 3D dynamique. Toutefois, la quantité de données peut être énorme. Des maillages de haute qualité peuvent être composés par des millions de sommets. Les canaux de transmission sont limités, et dans de nombreux cas coûteux. Les téléphones portables, par exemple, sont actuellement capables de rendus 3D et les applications mobiles 3D sont de plus en plus populaires. Pour la réussite de ces services et applications, impliquant dans certains cas la transmission sur les réseaux 3G, une compression efficace est cruciale. En outre, les capacités de stockage sont encore une ressource coûteuse.

Une première étape pour une compression efficace est d'utiliser des animations avec des images-clés. Au lieu de spécifier le maillage à chaque instant du temps, seulement une séquence de maillages importants (les maillages-clés) est définie explicitement. Les positions des sommets des maillages manquants, qui sont alors définies par des fonctions d'interpolation, peuvent correspondre à une interpolation linéaire dans le cas le plus simple. Ce type d'animations est largement utilisé pour la production d'animations 3D. L'opérateur conçoit les maillages-clés et la séquence complète est calculée par interpolation. Une telle représentation est déjà assez compacte. Cependant, elle peut encore être compressée de façon plus efficace. En effet, de nombreux travaux ont été développés et publiés dans la littérature dans le but de compresser ces séquences de maillages [21, 89, 60, 47, 9, 10, 5, 100, 23, 7, 75, 137, 138, 67, 126, 143, 122, 95, 62, 141, 142, 27, 28, 26].

Comme nous l'avons vu dans la première partie de cette thèse, l'irrégularité des maillages ne permet pas d'exploiter efficacement la corrélation spatiale. Par conséquent, la majorité de ces travaux exploitent la corrélation temporelle des séquences de maillages en utilisant, par exemple, des techniques de compensation en mouvement, des transformées en ondelettes temporelles et des transformées en cosinus discrètes.

Dans la compression géométrique, il a été démontré que la combinaison d'un remaillage semi-régulier et une transformée en ondelettes basées sur les maillages semi-réguliers conduit à des techniques efficaces pour la compression des formes statiques [79, 35, 112, 113, 86, 124, 77, 31]. Comme nous l'avons vu dans la première partie, les maillages semi-réguliers ont en effet, l'avantage de présenter une structure multirésolution qui facilite l'utilisation d'une transformée en ondelettes (pour exploiter la cohérence spatiale), et également la visualisation, la transmission progressive, etc.

Ainsi, nous proposons, dans cette partie, une nouvelle approche pour la compression de séquences de maillages surfaciques basée sur une analyse spatio-temporelle afin de prendre en compte la corrélation temporelle, mais aussi la corrélation spatiale. Pour cela, nous avons développé une technique de remaillage semi-régulier pour des séquences de maillages surfaciques (afin d'obtenir des séquences de maillages semi-réguliers). Par la suite, nous avons proposé une analyse spatio-temporelle basée ondelettes pour les séquences de maillages semi-réguliers. Enfin, nous avons proposé une allocation binaire

optimisée pour notre schéma d'analyse. Le schéma général de notre approche, qui permet la compression des séquences de maillages, est illustré figure 1.

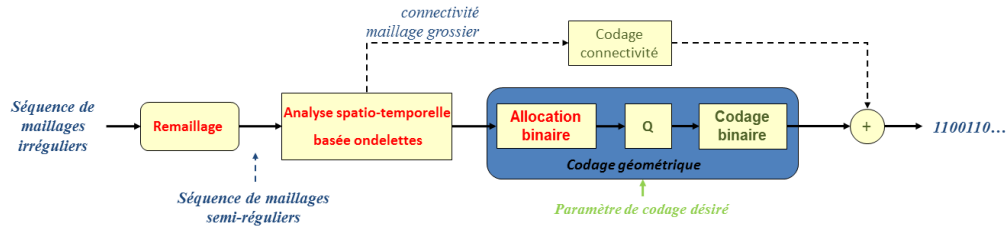


FIGURE 1 – Schéma de compression proposé pour les séquences de maillages surfaciques.

L'organisation de cette partie est la suivante : dans le premier chapitre, nous introduisons des notions générales sur les animations 3D et nous faisons un état de l'art sur la compression des séquences de maillages surfaciques. Dans le deuxième chapitre, nous détaillons notre contribution pour le remaillage semi-régulier des séquences de maillages surfaciques. Dans le troisième et dernier chapitre, nous détaillons notre analyse spatio-temporelle pour les séquences de maillages surfaciques semi-réguliers.

Chapitre 4

Modélisation et compression d'animations 3D

4.1 Introduction

Une illusion de mouvement est créée lorsque des images, qui sont légèrement différentes, sont affichées successivement. L'animation est le processus d'organisation et d'affichage des objets immobiles pour produire les images nécessaires afin de créer une telle illusion du mouvement. Les techniques d'animation peuvent être classées en deux principales catégories, à savoir : l'animation traditionnelle et l'animation par ordinateur.

Les films *Cartoon* sont l'exemple le plus répandu d'animation traditionnelle. Ils sont produits par une méthode appelée animation de celluloïds. Ce genre d'animation est assuré par les animateurs qui dessinent et peignent chaque image à la main. Les films *Cartoon* sont un secteur important de l'industrie du divertissement depuis les années 1930, ceci est une conséquence de la réussite des studios Walt Disney [109].

Les animations par ordinateur peuvent être divisées en deux groupes : animations assistées par ordinateur et animations générées par ordinateur [140]. Les animations assistées par ordinateur sont assimilées à des réalisations informatiques des animations de celluloïds 2D traditionnelles. En effet, le papier, la peinture, les pinceaux et les différents matériels de dessin sont remplacés par des ordinateurs, des scanners, des caméras, etc. L'ordinateur est principalement utilisé pour la peinture des celluloïds et l'interpolation. De cette façon, les dessins animés traditionnels peuvent être effectués plus efficacement et à moindre coût. Les animations générées par ordinateur sont également connues comme les "vraies" animations par ordinateur. Dans ce genre d'animations, les images sont générées par des moyens de rendu de modèles 3D. Ainsi, le mouvement est créé par la modification des modèles 3D au cours du temps. Ces modèles 3D ont différents

paramètres, tels que les positions des sommets, les angles des articulations, les valeurs de contraction musculaire, les couleurs et les paramètres de la caméra. Par conséquent, l'animation est créée en faisant varier les paramètres dans le temps et en effectuant le rendu des modèles pour générer le film d'animation final [140].

Le reste de ce chapitre est organisé comme suit : nous présentons, dans la section 4.2, les séquences de maillages surfaciques et le concept d'animation par images-clés. Ensuite, nous faisons, dans la section 1.4, un état de l'art sur la compression des séquences de maillages surfaciques. Enfin, nous donnons nos motivations et nos contributions dans la section 4.4.

4.2 Séquences de maillages surfaciques

Cette deuxième section présente des notions générales sur les séquences de maillages surfaciques et la méthode d'animation par images-clés, qui permet de créer des animations fluides.

4.2.1 Notions générales

Les séquences de maillages surfaciques sont très utilisées pour créer des animations 3D. Une séquence de maillages surfaciques est considérée comme un ensemble de maillages surfaciques $S_M = \{M_0, M_1, \dots, M_{N_M}\}$, qui représente un objet animé à différents instants (voir figure 2).

Chaque maillage de la séquence présente une information géométrique (l'ensemble des sommets des triangles) et une information de connectivité (l'ensemble des arêtes traduisant la manière avec laquelle sont connectés les sommets). D'autres informations peuvent s'ajouter à ces deux principales informations, à savoir : des textures, des couleurs pour améliorer la modélisation et le rendu de l'objet, des informations concernant la normale à la surface ou la courbure en chaque sommet, qui peuvent faciliter la visualisation des objets. Nous pouvons distinguer les deux catégories de séquences suivantes :

- les séquences à connectivité constante, où l'information de connectivité est la même pour les maillages constituant la séquence ;
- les séquences à connectivité variable, où chaque maillage de la séquence a sa propre connectivité et ses propres sommets. Dans ce cas, chaque maillage est donc indépendant.

Dans ce mémoire de thèse de doctorat, nous mettons l'accent uniquement sur les séquences à connectivité constante, qui sont les plus utilisées dans le cadre d'une animation par images-clés.

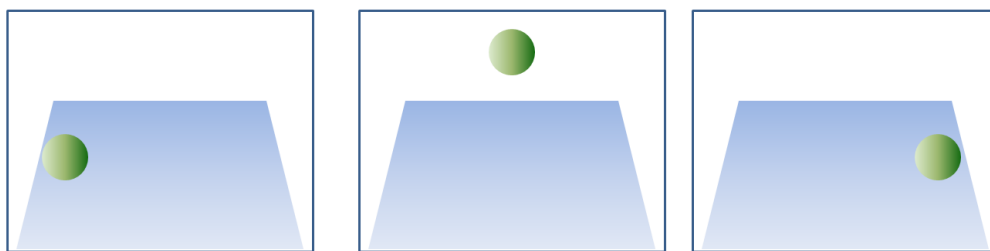
4.2.2 Animation par images-clés

L'animation par images-clés est la méthode la plus simple et la plus primitive pour créer des animations fluides. Nous fournissons à l'ordinateur une série de maillages à

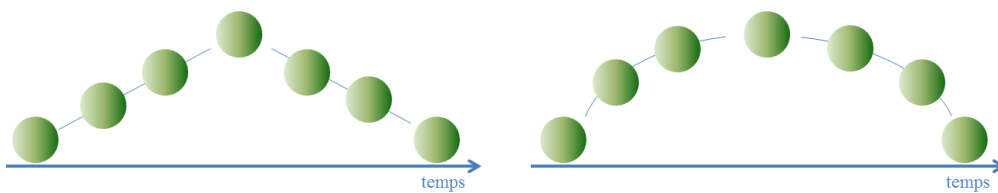


FIGURE 2 – Séquence de maillages surfaciques CHICKEN.

des temps donnés et l'ordinateur calcule les maillages intermédiaires par interpolation. Si l'interpolation utilisée est linéaire, alors l'objet se déplacera à vitesse constante et des discontinuités peuvent être observés lors de l'animation. Afin d'avoir un mouvement fluide, des techniques d'interpolation (B-spline, etc.) peuvent être utilisées. La figure 3 montre des exemples d'une telle animation.



(a) Séquence de maillages représentant un ballon rebondissant et composée par trois trames.



(b) A gauche, l'animation obtenue par interpolation linéaire. A droite l'animation obtenue par interpolation quadratique

FIGURE 3 – Exemples d'animations par images-clés créées à partir d'une séquence de maillages.

4.3 Etat de l'art

Les techniques de compression de séquences de maillages 3D peuvent être classées en cinq grandes familles :

- les techniques basées sur la segmentation du maillage de la séquence,
- les techniques exploitant une analyse en composantes principales (ACP),
- les techniques utilisant la prédiction spatio-temporelle locale,
- les techniques incluant une approche multirésolution (en particulier la transformée en ondelettes),
- les techniques utilisant les séquences de maillages semi-réguliers en entrée.

Un riche état de l'art a été proposé dans la thèse de Boulfani [21], où l'auteur a établi une synthèse bibliographique des principales techniques de compression de séquences de maillages 3D.

4.3.1 Compression basée segmentation du maillage de la séquence

La première méthode de compression de maillages animés a été proposée par Lengyel [89]. L'auteur propose un codeur qui exploite une segmentation basée mouvement du maillage de la séquence en plusieurs parties, suivi d'une compensation de mouvement. L'ensemble des matrices affines obtenues (décrivant le mouvement de chaque partie) est codé sans pertes, et les erreurs de compensation de mouvement sont quantifiées et codées avec un codeur arithmétique.

Depuis, plusieurs travaux ont vu le jour proposant une extension du codeur proposé par Lengyel, comme le codeur pour les séquences de maillages à connectivité constante proposé par Gupta *et al.* [60] et qui l'ont généralisé par la suite pour les séquences à connectivité variable dans le temps [59]. Une autre extension du codeur de Lengyel peut se trouver dans les travaux de Collins *et al.* [47] qui ont proposé une segmentation basée sur une minimisation au sens des moindres carrées, en regroupant les sommets de façon itérative jusqu'à ce que l'erreur globale de compensation de mouvement devienne supérieure à un seuil fixé. D'autres travaux proposant un codeur basé segmentation sont proposés dans les travaux de Amjoun *et al.* [9] et Amjoun et Straßer [10], où les auteurs ont établi une segmentation basée mouvement du maillage de la séquence basée sur le principe de croissance de région. Les auteurs ont proposé de coder les coordonnées des sommets par la transformée en cosinus discrète (TCD) temporelle, puis une quantification uniforme et un codage arithmétique des coefficients de la TCD.

Enfin, nous citons les travaux de Ahn *et al.* [5], où les auteurs proposent une segmentation basée sur la connectivité constante du maillage, consistant à décomposer le maillage en bandes de triangles de longueur quasi-uniforme. Ces bandes sont après compensées en mouvement par un vecteur vitesse moyen. Puis, une TCD 1D spatiale est appliquée sur les erreurs de compensation de mouvement séparément sur les coordonnées x , y et z des sommets.

Les travaux qui ont attiré aussi notre attention sont ceux établis par Mamou et al., résumés dans [100]. L'auteur a proposé deux schémas de compression, appelés MCGV (*Multi-Chart Geometry Video*) et FAMC (*Frame-based Animated Mesh Compression*). Les deux approches incluent : une technique de segmentation du maillage de la séquence au sens du mouvement fondée sur une représentation hiérarchique et une compensation de mouvement basée sur le modèle de peau (*skinning*), inspiré des techniques d'animation 3D qui permet de modéliser de manière fidèle le champ de mouvement, tout en s'affranchissant des problèmes de discontinuités au niveau des frontières. En premier lieu, les auteurs ont proposé le codeur MCGV, une approche fondée sur une représentation par images géométriques inspirée des travaux de Briceno *et al.* [23]. Ils ont proposé une représentation de la géométrie du maillage dynamique sous forme d'une séquence vidéo d'images géométriques 2D, compressée par les méthodes standards de codage d'images (JPEG ou MPEG-4 V2 [48]). Contrairement à l'approche proposée dans [23], le codeur MCGV n'inclue aucune procédure de remaillage. Il ne présente donc pas de discontinuités et préserve la connectivité initiale du maillage, qui est codée sans perte. De plus, ils améliorent considérablement les performances de compression proposée dans [23], tout en gardant les fonctionnalités offertes par les standards de compression, comme la scalabilité temporelle, la scalabilité en qualité, *streaming* et codage des attributs. Toutefois, le codeur MCGV n'offre pas les fonctionnalités de scalabilité spatiale et de codage quasi-sans perte.

Pour cela, les auteurs proposent le codeur FAMC [103, 104] qui englobe plusieurs approches de compression en se basant sur différentes techniques comme : la TCD appliquée temporellement avec une prédiction delta des coefficients, la prédiction hiérarchique LD (*Layered Decomposition*) et enfin l'analyse en composantes principales (ACP). Ainsi, ils ont pu augmenter l'efficacité de la compression par rapport à l'approche MCGV, mais aussi proposer une technique de codage générale avec une architecture flexible et modulaire supportant toutes les fonctionnalités requises, comme le *streaming*, la scalabilité (temporelle, spatiale et en qualité), la prise en compte du codage des attributs photométriques et de compression quasi-sans perte. Les hautes performances qu'offre le codeur FAMC lui ont permis d'intégrer le standard ISO/MPEG-4 [101, 102].

4.3.2 Compression basée sur une analyse en composantes principales

Les premiers à avoir proposé un codeur basé ACP pour les séquences de maillages sont Alexa et Müller [7]. Ils ont établi une compensation de mouvement rigide avec une décomposition ACP du champ de mouvement résiduel. Ainsi, la géométrie du maillage animé est représentée avec un petit nombre de fonctions de la base ACP. Depuis, une multitude de travaux ont été proposés incluant l'approche ACP. En effet, Karni et Gotsman ont amélioré la technique proposée dans [7] en exploitant les corrélations spatiale et temporelle. Ils ont proposé de coder les coefficients ACP avec un schéma de codage prédictif appelé LPC (*Linear Prediction Coding*) [75].

Les codeurs proposés par Alexa et Müller [7] et par Karni et Gotsman [75] sont efficaces que pour les séquences avec un mouvement périodique présentant un grand nombre de trames par rapport aux nombre de sommets. Mais aussi, ils présentent une grande complexité de calcul due au traitement spatial et temporel de la séquence. De ce fait, Vasa et Skala ont proposé une amélioration du codeur présenté dans [75] en l'optimisant pour les séquences présentant des maillages fins (un grand nombre de sommets) à forte corrélation spatiale [137]. Ils ont également réduit la complexité du calcul en proposant une méthode de segmentation en divisant la séquence en sous groupes de trames traités indépendamment chacun, mais aussi en appliquant l'ACP de façon indépendante sur chaque partie de la partition du maillage au lieu de le faire globalement sur l'ensemble du maillage. Par la suite, ils ont aussi proposé d'associer l'approche ACP à des prédicteurs géométriques LSP (Least Squares Prediction) et RBFP (Radial Basis Function-based Predictor) dans [138].

Sattler *et al.* ont proposé un codeur appelé CPCA (*Clustered Principle Component Analysis*) qui inclut une technique de segmentation non automatique, avec une ACP appliquée sur les trajectoires des sommets de chaque partie [118].

4.3.3 Compression basée sur une prédiction spatio-temporelle locale

Ibirra et Rossignac ont proposé une méthode de compression de maillages à connectivité constante appelée *Dynapack*, dont le principe est de prédire la position d'un sommet en utilisant le voisinage spatial et la position du sommet déjà décodé dans la trame précédente de la séquence de maillages [67].

Dans le même contexte, Stefanoski et Ostermann ont proposé un prédicteur optimal pour les mouvement rigides, dont le principe consiste à projeter chaque sommet du maillage dans un repère orthogonal construit autour de lui [126]. Yang *et al.* ont proposé un autre prédicteur à faible complexité de calcul, consistant à prédire le mouvement des sommets en fonction de la moyenne pondérée de ses sommets voisins décodés [143].

Briceno *et al.* ont proposé une approche originale appelée GV (*Geometry Video*), consistant à projeter chaque trame de la séquence en une image 2D [23]. Une compensation de mouvement des image 2D est proposée, et les erreurs résiduelles de la compensation sont encodées avec un codeur vidéo 2D EZW (*Embedded Zerotree Wavelet encoder*).

4.3.4 Compression basée sur des techniques multirésolutions

Shamir et Pascucci ont proposé un codeur basé sur une décomposition multirésolution [122]. Ils calculent les transformations affines des maillages animés par rapport à la première trame. Les résiduels du mouvement d'une trame par rapport à la trame précédente, compensée par les transformées affines sont alors encodés par un codeur basé sur une structure multirésolution appelé TDAG (*Temporal Directed Acyclic Graph*).

Lopes et Gamito ont proposé une technique de compression basée sur la transformée en ondelettes temporelle [95]. Plus précisément, une transformée en ondelettes B-spline cubique 1D est appliquée sur les champs de vitesses de l'animation, séparément selon les axes X , Y et Z . Les coefficients de la transformée on ondelettes supérieures à un seuil fixé sont codés, le reste est mis à zéro.

Guskov et Khodakovsky ont proposé le codeur AWC (Animation Wavelet Compression) qui est basé sur l'analyse multirésolution spatiale [62]. Dans leurs travaux, l'analyse est appliquée sur la géométrie de chaque trame de la séquence afin d'obtenir une représentation multirésolution. Un codage prédictif basé sur l'approche *I-frames/P-frames*, utilisée dans le codage vidéo (voir figure 4), est appliqué sur les détails résultants de chaque trame.

Cho *et al.* ont proposé une méthode de compression basée ondelettes quasi-sans pertes pour les séquences de maillages à connectivité constante permettant d'exploiter la redondance spatiale et temporelle des maillages [27]. En s'inspirant de l'approche développée dans MPEG2 La séquence est représentée par la structure *I-frames/P-frames/B-frames* (voir figure 4). La technique inclut une transformée en ondelettes spatiale irrégulière adaptative basée sur les ondelettes paresseuses (*lazy wavelet*) et développée dans [136] pour le cas des maillages statiques. Un codage par prédiction temporelle est proposée pour coder les coefficients des ondelettes des I-frames et B-frames. Quant aux I-frames, elles sont codées avec le codeur statique proposé dans [136].

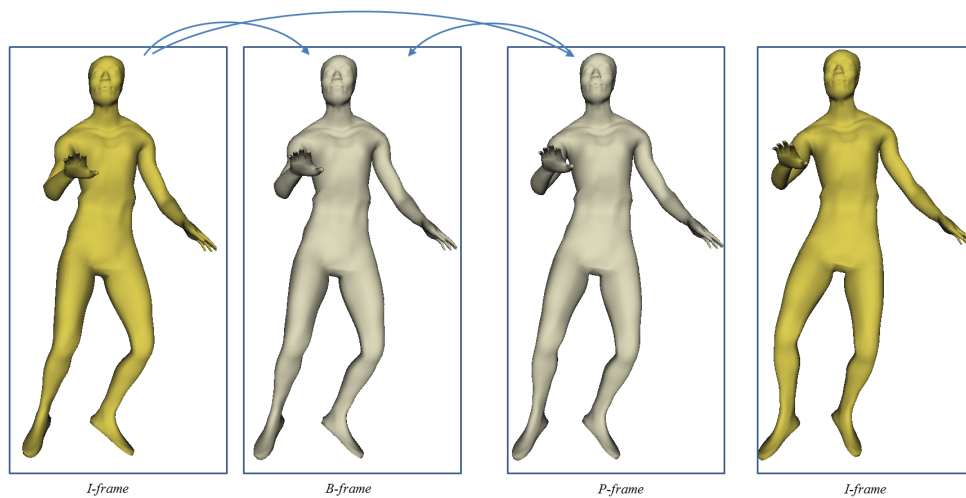


FIGURE 4 – Exemple de structure avec des *I-frames/P-frames/B-frames*. Une *I-frame* est une trame codée indépendamment des autres trames. Une *P-frame* est une trame prédite à partir d'une trame qui la précède. Une *B-frame* est une trame prédite à partir d'une trame qui la précède et d'une trame qui la suit.

Par la suite, Cho *et al.* ont amélioré les performances de leur codeur en proposant le codeur SWA+TWA (*Spatial Wavelet Analysis et Temporal Wavelet Analysis*) [28]. Ce co-

deur inclut une transformée en ondelettes temporelle associée à la transformée en ondelettes spatiale irrégulière et le codage différentiel temporel de premier ordre *FDC* (First Differential Coding) proposé précédemment dans leurs anciens travaux [27]. L'analyse par la transformée en ondelettes temporelle a été assurée par trois bancs de filtres : Haar, Le Gall et Daubechies. Récemment, ils ont encore amélioré les performances de leurs codeurs, en proposant le codage différentiel temporel *MDC* (*Multi-order Differential Coding*) [26].

Nous citons enfin, les travaux de thèse de Boulfani [21] qui a proposé un schéma de compression des séquences de maillages basé sur une compensation de mouvement, suivie d'une transformée en ondelettes temporelle.

4.3.5 Compression basée sur un remaillage semi-régulier

À notre connaissance, il existe une seule technique pour le remaillage semi-régulier des séquences de maillages à connectivité constante. Cette technique a été proposée par Yang *et al.* dans [142]. Ils ont proposé de construire une structure semi-régulière pour le premier maillage de la séquence et par la suite l'adapter aux autres maillages de la séquence en se basant sur une estimation de mouvement. Par la suite, ils ont proposé une technique de compression utilisant le schéma *lifting* basé Butterfly [141].

4.4 Motivations et contributions

Nous avons vu qu'il y a beaucoup de travaux qui ont été développés pour la compression de ces séquences. La plupart de ces travaux n'exploitent pas la corrélation spatiale et se limitent à diminuer la redondance temporelle. Ceci peut s'expliquer par l'échantillonnage irrégulier des maillages.

En effet, très peu de travaux exploitent les corrélations temporelle et spatiale. Parmi ces travaux, on distingue notamment ceux de Cho *et al.* [27] qui utilise une transformée en ondelettes spatiale irrégulière et les travaux de Yang *et al.* [141], qui ont proposé une technique de remaillage semi-régulier des séquences de maillages surfaciques afin d'utiliser efficacement une transformée en ondelette spatiale. Inspirés par leurs travaux, nous proposons dans cette partie notre schéma de compression pour les séquences de maillages surfaciques.

Dans le prochain chapitre, nous présentons notre première contribution pour les maillages dynamiques, à savoir notre approche pour le remaillage semi-régulier, une étape nécessaire pour obtenir des séquences de maillages semi-régulier. Dans le chapitre 6, nous présentons notre deuxième contribution, à savoir notre schéma de compression pour les séquences ainsi obtenues. Notre schéma inclut une analyse spatio-temporelle qui comporte non seulement une transformée en ondelettes spatiale, mais aussi une transformée en ondelettes temporelle. Nous proposons également une allocation binaire qui permettra de quantifier de manière optimale les sous-bandes de coefficients d'ondelettes, afin d'obtenir la meilleure qualité visuelle pour un débit donné.

Chapitre 5

Remaillage semi-régulier de séquences de maillages surfaciques

5.1 Introduction

Pour les maillages statiques, le problème de l'échantillonnage irrégulier a été surmonté grâce à des techniques de remaillage semi-réguliers [97, 46, 56, 64, 87, 61, 88, 53, 6, 83, 70], qui permettent de transformer les maillages irréguliers en des maillages semi-réguliers. Ces maillages semi-réguliers ont un grand avantage en comparaison aux maillages irréguliers. D'une part, ils ont une structure multirésolution implicite qui permet, par exemple, d'afficher facilement le maillage avec différents niveaux de détails. D'autre part, ils ont un échantillonnage plus régulier qui permet d'utiliser efficacement des transformées en ondelettes, et par conséquent d'améliorer les taux de compression [79, 35, 112, 113, 86, 124, 77, 31].

Dans ce chapitre, nous proposons notre technique de remaillage semi-régulier de séquences de maillages, et ce, en s'inspirant des travaux développés pour les maillages statiques et ceux proposés par Yang *et al.* concernant le remaillage semi-régulier des séquences de maillages surfaciques [142]. Cette étape de remaillage nous permettra par la suite d'exploiter la corrélation spatiale en plus de la corrélation temporelle, et ce, en combinant une transformée en ondelettes spatiale à une transformée en ondelettes temporelle (voir chapitre 6).

Ce chapitre est organisé de la manière suivante : nous présentons globalement notre remailleur semi-régulier de séquences de maillages surfaciques dans la section 5.2. Ensuite, nous détaillons les différentes étapes de notre approche dans la section 5.3. Par la suite, nous donnons des résultats expérimentaux et visuels dans la section 5.4. Enfin, nous concluons dans la section 5.5.

5.2 Présentation de la méthode proposée

Nous présentons dans cette section notre approche pour le remaillage semi-régulier de séquences de maillages surfaciques à connectivité constante. La figure 5.1 montre le principe de notre approche, qui se résume ainsi :

- choix d’un maillage de référence M_{ref} ;
- segmentation basée mouvement du maillage. Cette segmentation permet d’obtenir un ensemble de régions (ou *clusters*) contenant chacun des sommets ayant un mouvement similaire. Le mouvement des sommets de chaque partition est approximé par une matrice affine ;
- recalage de tous les maillages qui constituent la séquence sur le maillage de référence M_{ref} ;
- calcul du maillage moyen des maillages recalés et du maillage de référence M_{ref} ;
- remaillage de ce maillage moyen par un remailleur semblable à MAPS [87] ;
- déformation du maillage semi-régulier, ainsi obtenu, avec les matrices affines caractérisant le mouvement de l’objet 3D pour reconstruire la séquence.

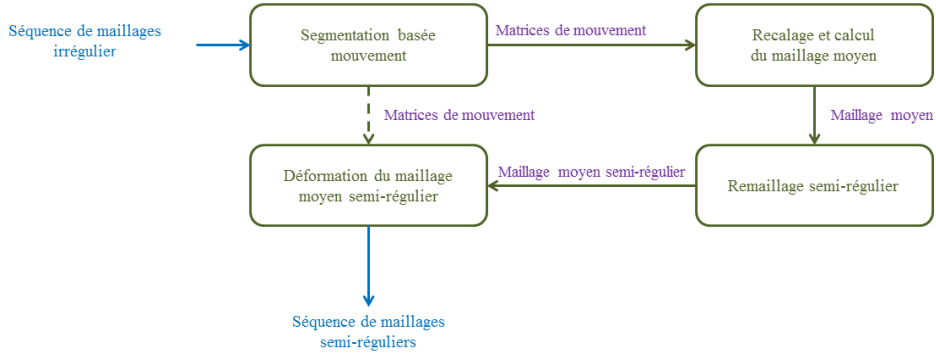


FIGURE 5.1 – Principe de notre remailleur semi-régulier pour les séquences de maillages surfaciques.

5.3 Détails de l’approche proposée

Dans ce qui suit, nous détaillons les différentes étapes de l’approche que nous avons proposée pour le remaillage semi-régulier d’une telle séquence.

5.3.1 Segmentation basée mouvement

Soit une séquence de maillages surfaciques à connectivité constante $\mathcal{S}_M = \{M_0, M_1, \dots, M_T\}$ et un maillage de référence M_{ref} choisi arbitrairement.

Nous devons commencer par segmenter le maillage en un ensemble de régions connexes $R_C = \{C_0, C_1, \dots, C_{N_C}\}$ et estimer le mouvement de toutes ces régions, en

vue de remailler la séquence S_M . Pour cela, nous utilisons la segmentation basée mouvement proposée dans [22], qui consiste à regrouper dans la même région l'ensemble des sommets ayant un mouvement similaire. Dans ce cas, deux maillages M_i et M_j de la séquence S_M doivent être choisis.

Ensuite, nous estimons le mouvement de tous les sommets v^k du maillage par une transformée affine, qui est caractérisée par la matrice A^k suivante :

$$A^k = \begin{pmatrix} [R^k]_{3 \times 3} & [T^k]_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (5.1)$$

avec R^k la matrice définissant la rotation, et T^k le vecteur caractérisant la translation du sommet v^k .

Afin d'estimer cette matrice affine, nous prenons en compte le voisinage d'ordre 1 du sommet v^k , qui représente l'ensemble des sommets $\{\bar{v}_i^{k,n}\}$ qui lui sont liés par une arête. Ainsi, nous définissons les matrices \mathcal{P}_i^k et \mathcal{P}_j^k qui contiennent ces sommets et leurs positions dans les maillages M_i et M_j , respectivement :

$$\mathcal{P}_i^k = (v_i^k, \bar{v}_i^{k,1}, \bar{v}_i^{k,2}, \dots, \bar{v}_i^{k,N^k}), \quad (5.2)$$

avec N^k le nombre de voisins du sommet v^k . Chaque sommet de ces matrices est définie par ces trois coordonnées.

Une fois les matrices \mathcal{P}_i^k et \mathcal{P}_j^k sont définies, nous calculons la matrice affine A^k telle que :

$$A^k = \mathcal{P}_j^k (\mathcal{P}_i^k)^+, \quad (5.3)$$

où l'opérateur $(.)^+$ est le pseudo-inverse d'une matrice. le pseudo-inverse d'une matrice A est calculé à partir de sa décomposition en valeurs singulières. Soit $A = USV^T$, avec S la matrice contenant les valeurs singulières, et U et V deux matrices orthogonales. Nous pouvons alors écrire : $A^+ = VS^+U^T$, où la matrice S^+ est calculée en remplaçant dans S toutes les valeurs singulières par leur inverse.

Nous calculons ainsi toutes les matrices affines caractérisant les mouvements de tous les sommets et, par la suite, nous regroupons, dans la même région, les sommets ayant un mouvement similaire. Pour cela, nous segmentons l'ensemble des sommets en appliquant l'algorithme de Lloyd [93], déjà présenté dans le deuxième chapitre de la première partie, sur les matrices affines calculées. Un exemple de cette segmentation est donnée figure 5.2.

5.3.2 Recalage et calcul du maillage moyen

En partant de la segmentation calculée dans la sous-section précédente, nous recalons tous les maillages de la séquence S_M sur le maillage de référence M_{ref} . Pour cela, nous estimons les mouvements de chaque région C^t à l'aide de matrices affines Mv_i^t qui caractérisent le mouvement des sommets appartenant à C^t et pour tous les maillages $\{M_i\}$

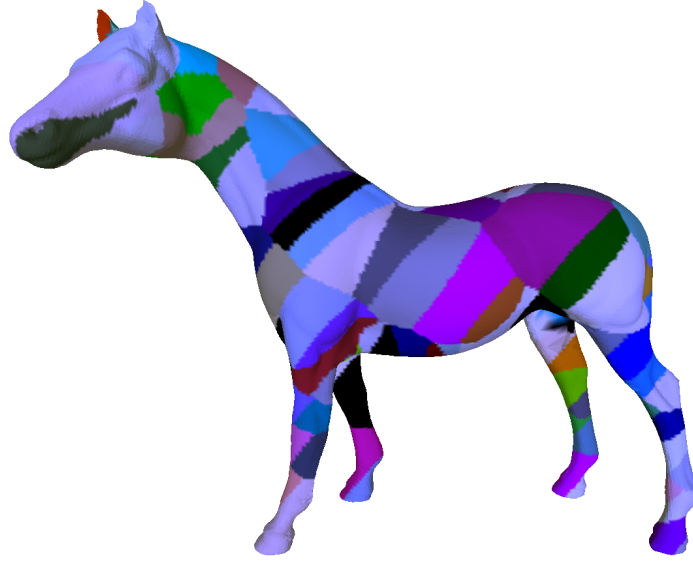


FIGURE 5.2 – Segmentation de la séquence FLYING HORSE.

de la séquence S_M . Pour un maillage M_i de la séquence, cette matrice Mvt_i^t est donnée par l'expression suivante :

$$Mvt_i^t = \mathcal{V}_{ref}^t * (\mathcal{V}_i^t)^+, \quad (5.4)$$

où \mathcal{V}_i^t et \mathcal{V}_{ref}^t sont les matrices contenant les positions des sommets appartenant à la région C_t dans le maillage M_i et le maillage M_{ref} , respectivement. Ces matrices sont définies par l'expression suivante :

$$\mathcal{V}_i^t = (v_i^{t1}, v_i^{t2}, \dots, v_i^{tN^t}), \quad (5.5)$$

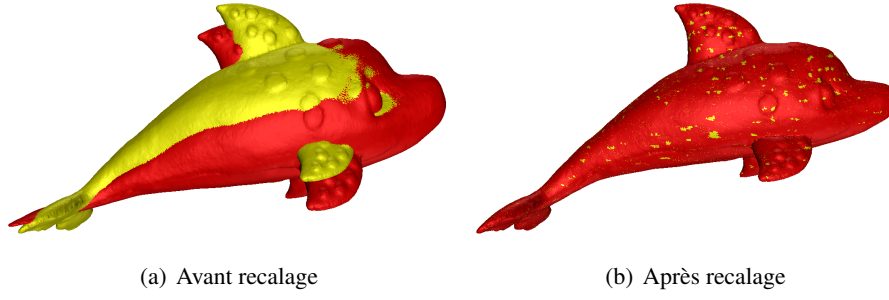
où N^t représente le nombre de sommets de C^t .

Une fois ces matrices calculées pour toutes les régions et tous les maillages de la séquence, nous devons recalcr chaque maillage M_i sur le maillage de référence M_{ref} . Ainsi, nous recalons toutes les régions de M_i sur les régions de M_{ref} , soient :

$$\begin{aligned} \tilde{C}_i^0 &= Mvt_i^0 C_{ref}^0 \\ \tilde{C}_i^1 &= Mvt_i^1 C_{ref}^1 \\ &\vdots \\ \tilde{C}_i^{N_C} &= Mvt_i^{N_C} C_{ref}^{N_C}, \end{aligned} \quad (5.6)$$

où \tilde{C}_i^k représente la région C^k du maillage M_i recalée sur la même région du maillage M_{ref} .

La figure 5.3 illustre le recalage d'un maillage de la séquence DOLPHIN.

FIGURE 5.3 – Recalage du maillage M_{44} sur le maillage M_1 .

Une fois le recalage terminé, nous calculons un maillage moyen M_{moy} , qui a la même connectivité des maillages de la séquence. Chaque sommet $v_{moy}^i \in M_{moy}$ est calculé comme suit :

$$v_{moy}^i = \sum_{j=0}^{j \leq N_M} v_i^j. \quad (5.7)$$

5.3.3 Remaillage du maillage moyen

Nous remaillons le maillage moyen avec un algorithme de remaillage pouvant être utilisé pour les maillages statiques. Le remailleur statique utilisé dans notre implémentation est basé sur MAPS [87] ; il diffère de ce dernier par l'étape de simplification. En effet, nous avons opté pour une méthode de simplification, qui est basée sur des décimations d'arêtes, tandis que la simplification dans MAPS est basée sur des décimations de sommets. Ce remailleur comporte les trois étapes suivantes :

- simplification,
- paramétrisation,
- subdivision.

Simplification

Nous avons utilisé l'algorithme de Garland *et al.* [55] basé sur la contraction de demi-arêtes, et soumis à quelques contraintes ajoutées pour l'adapter à nos besoins. Cet algorithme se résume comme suit :

- chaque triangle du maillage appartient à un plan $p = [abcd]$, d'équation $ax + by + cz + d = 0$, où $a^2 + b^2 + c^2 = 1$;
- associer à chaque triangle une matrice $K_p = pp^T$;
- associer à chaque sommet v_i la matrice Q_i suivante : $Q_i = \sum_{p \in \text{planes}(v_i)} K_p$;
- associer à chaque arête (v_1, v_2) un coût de contraction $\Delta = v_1^T Q v_1$, où $Q = Q_1 + Q_2$;
- placer les différentes arêtes constituant le maillage dans une queue de priorité ordonnée en fonction des coûts de contraction (du plus faible au plus grand) et

contracter une à une les arêtes jusqu'à obtenir le nombre de triangles souhaité.

La contraction d'une arête (v_1, v_2) est interdite dans les situations suivantes :

- si la contraction entraîne un changement topologique (le maillage devient "non-manifold"), voir [37]. Pour cela, il faut que v_1 et v_2 vérifient la condition suivante : $Link(v_1) \cap Link(v_2) = Link(v_1v_2)$, où $Link(v_i) =$ voisins de v_i , et les arêtes autour de v_i qui forment avec v_i des triangles du maillages et $Link(v_1v_2) =$ ensemble des sommets, avec qui l'arête forme des triangles du maillage (voir figure 5.4) ;
- si la contraction entraîne une inversion des normales [54]. Dans chaque triangle autour de v_2 , excepté ceux en communs avec v_1 , il existe une arête opposée à v_2 . Si nous faisons passer un plan perpendiculaire au triangle et passant par l'arête, alors il faut que v_1 et v_2 soient du même coté par rapport à ce plan (voir figure 5.5) ;
- si les normales à la surface changent, après la contraction, d'un angle supérieur à un certain seuil ;
- si la contraction aboutit à des triangles compacts (formés par des sommets presque colinéaires), voir [54]. Guéziec [65] a suggéré une heuristique pour définir la compacité d'un triangle. Elle est définie par : $\gamma = \frac{4\sqrt{3}w}{l_1^2 + l_2^2 + l_3^2}$, où les l_i sont les longueurs des arêtes et w est la surface du triangle (voir figure 5.6).

Un exemple d'objet simplifié est donné figure 5.7.

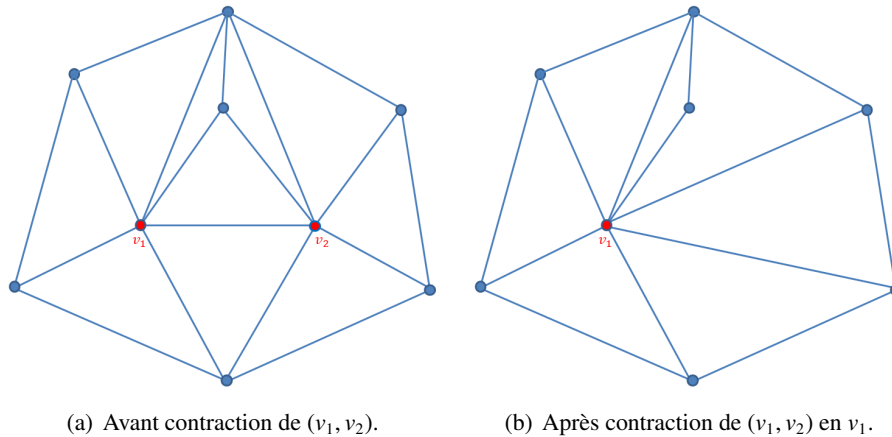


FIGURE 5.4 – Exemple de contraction entraînant un changement topologique.

Paramétrisation

Cette étape est effectuée simultanément avec l'étape de simplification. Pour chaque arête (v_n, v_i) , qui va être contractée en v_n , nous énumérons les K_i voisins $V(v_i) = \{v_{j_k} / 1 \leq k \leq K_i\}$ de v_i (le sommet qui va être supprimé), tels que chaque triplet $\{v_{j_{k-1}}, v_i, v_{j_k}\}$ forme un triangle du maillage et que $(j_0 = j_{K_i})$. Nous allons maintenant plaquer le voisinage de v_i sur un plan 2D et effectuer la contraction dans ce domaine planaire. Pour cela, nous

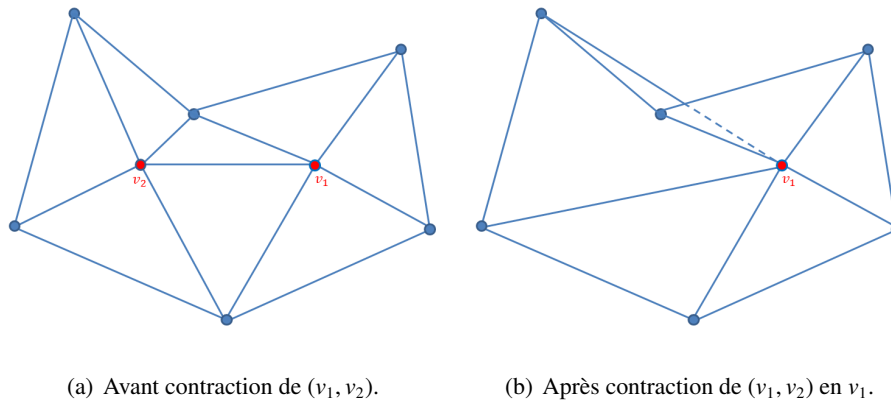


FIGURE 5.5 – Exemple de contraction entraînant une inversion de la normale.

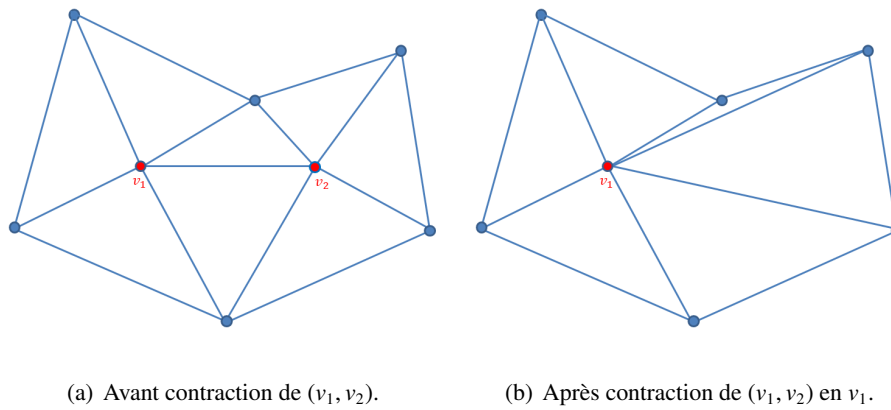


FIGURE 5.6 – Exemple de contraction entraînant un triangle compacte.

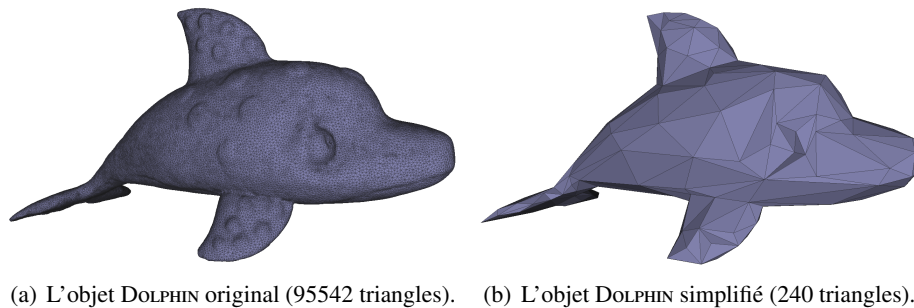


FIGURE 5.7 – L'objet DOLPHIN original et simplifié.

construisons la fonction de *mapping* μ_i , tels que : $\mu(v_i) = 0$ et $\mu(v_{j_k}) = r_k^a \exp(i\theta_k^a)$, où $r_k = \|v_i - v_{j_k}\|$,

$$\theta_k = \sum_{l=1}^k \arg(v_{j_{l-1}}, v_i, v_{j_l})$$

et $a = \frac{2\pi}{\theta_{K_i}}$, qui est une fonction linéaire par morceaux approximant la transformation conforme z^a définie dans [87, 43]. Notons qu'une transformation conforme est une fonction qui préserve les angles.

Ainsi, nous avons construit une paramétrisation, en vue de représenter le voisinage d'un sommet qui va être supprimé dans un plan 2D. Dans ce domaine planaire, nous contractons l'arête et nous supprimons le sommet v_i et les triangles contenant l'arête (v_n, v_i) , ce qui va changer la triangulation au voisinage du sommet v_i . Dans cette nouvelle triangulation, nous cherchons le triangle (v_k, v_l, v_m) , auquel appartient le sommet v_i , et nous écrivons : $v_i = \alpha v_k + \beta v_l + \gamma v_m$, où α, β et γ sont les coordonnées barycentriques de v_i dans ce triangle. Nous cherchons aussi pour tous les sommets déjà supprimés qui sont projetés dans un triangle appartenant au voisinage de v_i , les nouveaux triangles auxquels ils appartiennent.

De cette façon, nous construisons une bijection π entre le maillage original et le maillage simplifié. Il faut noter que l'image d'un triangle par cette bijection peut être un triangle avec une normale inversée. Pour remédier à ce problème, nous ajustons les paramètres de chaque sommet appartenant à un tel triangle, en les remplaçant par la moyenne de paramètres de ces voisins.

Subdivision

Nous devons maintenant subdiviser le maillage simplifié. Pour chaque sommet v_i appartenant au maillage simplifié, nous projetons son voisinage dans le plan 2D, en utilisant l'approximation de la transformation conforme, et nous subdivisons les triangles dans ce voisinage.

En partant de v_i , et en suivant un chemin en "zig-zag", nous essayons de trouver pour chaque nouveau sommet v_s le triangle (v_j, v_k, v_l) du maillage original auquel il appartient, en utilisant l'algorithme de Brown et Faigle [24]. Ensuite, nous écrivons l'expression suivante : $v_s = \alpha v_j + \beta v_k + \gamma v_l$, où α, β et γ sont les coordonnées barycentriques de v_s dans ce triangle.

5.3.4 Remaillage de la séquence

Une fois le maillage moyen M_{moy} remaillé, nous recalons chaque sommet v sur chaque maillage M^t comme suit :

- si v appartient au maillage moyen irrégulier, et donc n'a pas été supprimé lors de l'étape de simplification, alors il appartient à une certaine région C_k^i . Nous le recalons donc à l'aide de la matrice affine caractérisant le mouvement de la région C_k^i ;
- si v n'appartient pas au maillage moyen irrégulier, et donc a été supprimé lors de l'étape de simplification, alors il appartient à un triangle du maillage moyen original (v_m, v_n, v_o) , tel que : $v = \alpha v_m + \beta v_n + \gamma v_o$, où α, β et γ sont les coordonnées barycentriques de v dans (v_m, v_n, v_o) . Nous faisons une interpolation linéaire du mouvement à l'intérieur du triangle à partir des mouvements de ces trois sommets, qui peuvent le constituer. Nous recalons v à l'aide de la matrice affine Mvt , telle que : $Mvt = \alpha Mvt_m + \beta Mvt_n + \gamma Mvt_o$, avec Mvt_m, Mvt_n et Mvt_o des matrices affines caractérisant le mouvement des sommets v_m, v_n et v_o , respectivement.

5.4 Résultats expérimentaux

Afin de tester le remailleur dynamique que nous venons de décrire, nous avons créé trois séquences de maillages suffisamment denses pour que le remailage semi-régulier ait un intérêt. Ces séquences se nomment DOLPHIN, FLYING HORSE et FLYING VENUS et leurs principales caractéristiques sont présentées dans le tableau 5.1. Pour les tests qui suivent, nous avons choisi le premier maillage comme maillage de référence et nous avons utilisé les deux premiers maillages pour la segmentation.

TABLE 5.1 – Caractéristiques des séquences utilisées durant les expérimentations.

Séquence	Nombre de sommets	Nombre de triangles	Nombre de maillages
DOLPHIN	47773	95542	101
FLYING HORSE	48485	96966	100
FLYING VENUS	50002	100000	85

Les figures 5.8, 5.9 et 5.10 montrent les versions remaillées des trois séquences précédemment introduites, et ce à différents niveaux de détails. A titre indicatif, le tableau 5.2 donne le temps de calcul nécessaire pour le remailage de la séquence DOLPHIN. Nous remarquons que notre approche est relativement rapide.

5.5 Conclusion

Dans ce chapitre, nous avons présenté notre remailleur semi-régulier pour les séquences de maillages surfaciques. Cette approche nous permet d'obtenir des séquences de maillages semi-réguliers qui ont un échantillonnage plus régulier et une structure multi-résolution implicite. Ceci nous permettra par la suite (voir chapitre 6) d'appliquer une transformée en ondelettes spatiale en plus de la transformée en ondelettes temporelle, afin d'exploiter les corrélations temporelle et spatiale.

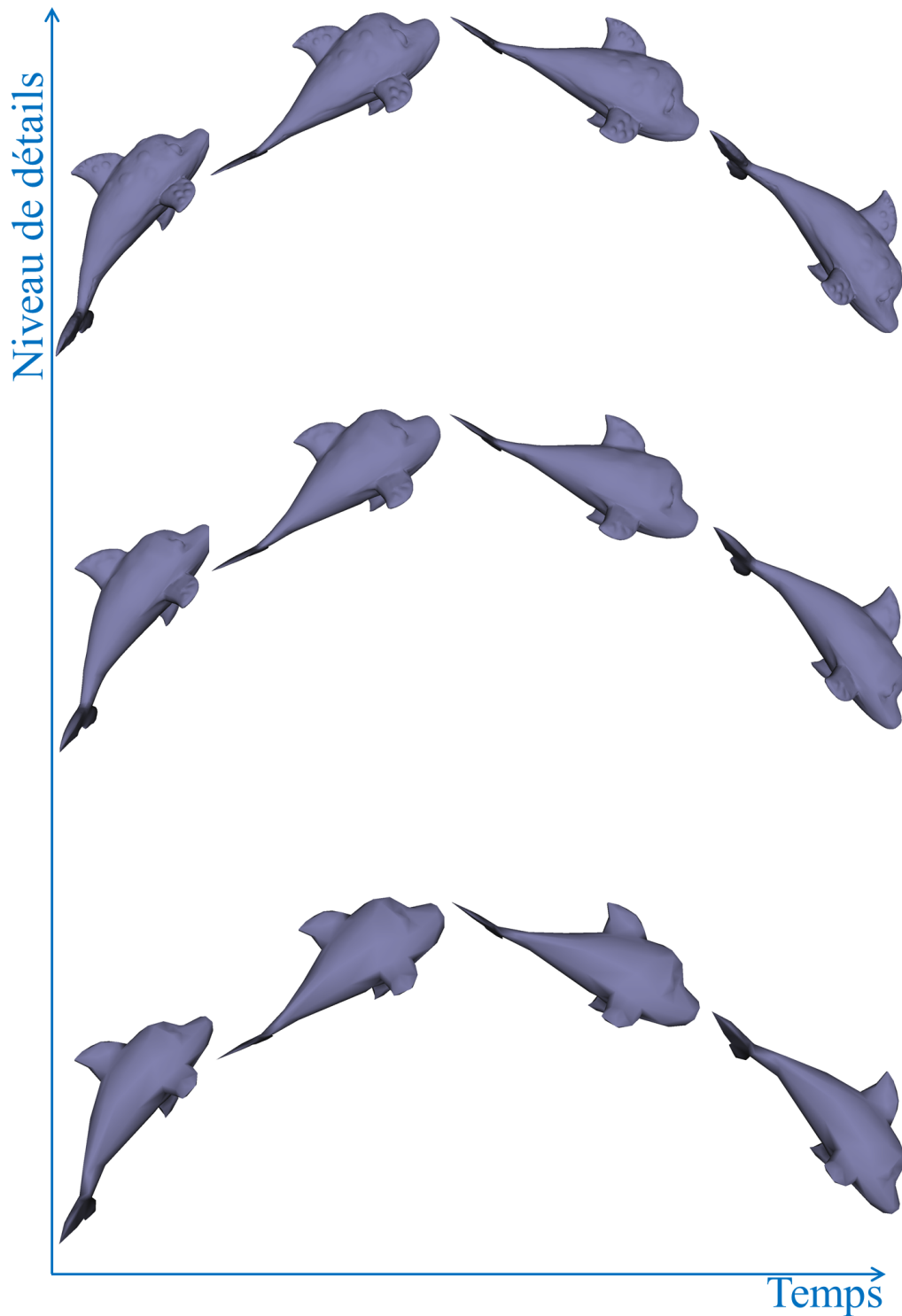


FIGURE 5.8 – En haut, la séquence DOLPHIN remaillée. Au milieu, la séquence DOLPHIN remaillée et représentée avec trois niveaux de détails. En bas, la séquence DOLPHIN remaillée et représentée avec un seul niveau de détails.

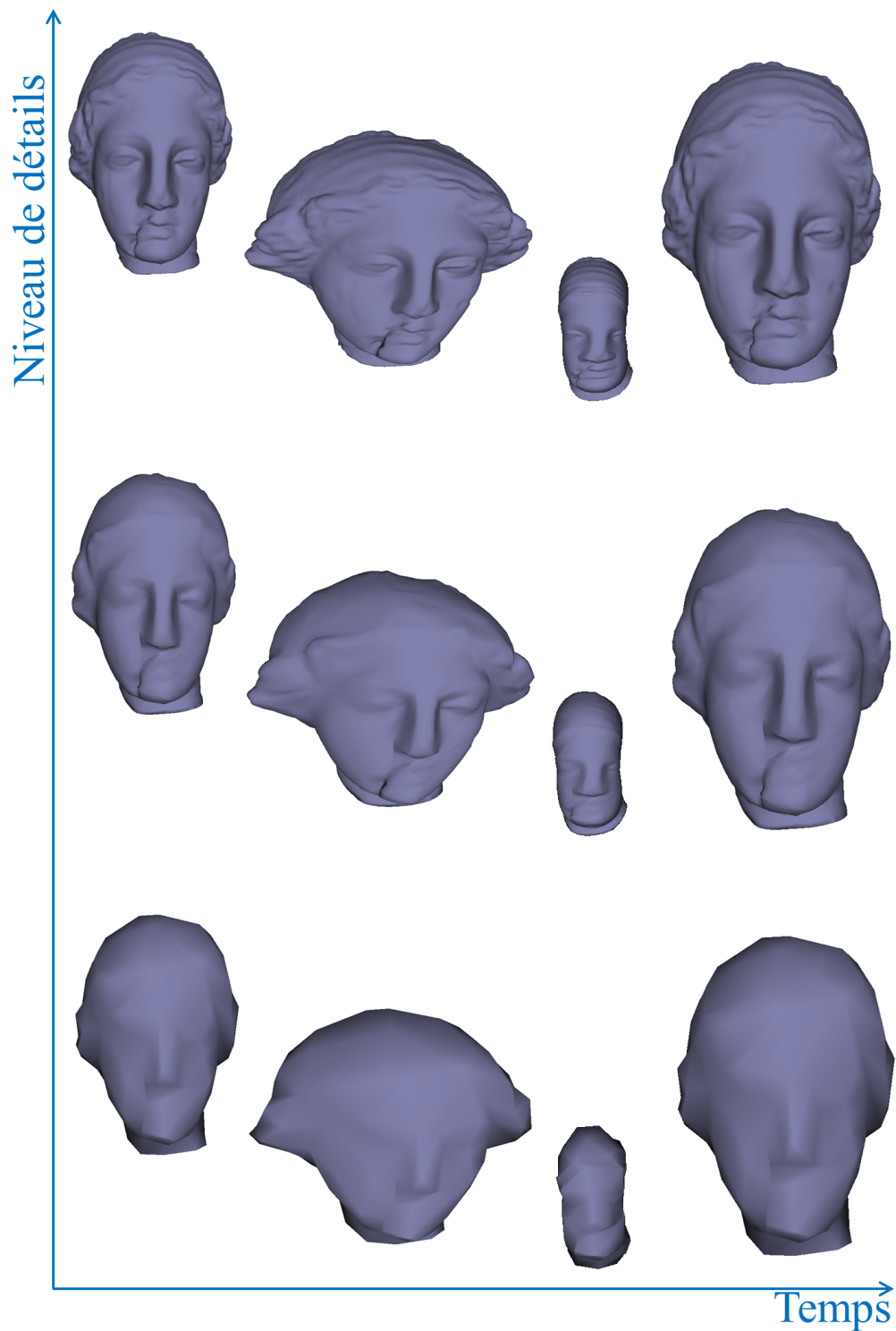


FIGURE 5.9 – En haut, la séquence FLYING VENUS remaillée. Au milieu, la séquence FLYING VENUS remaillée et représentée avec trois niveaux de détails. En bas, la séquence FLYING VENUS remaillée et représentée avec un seul niveau de détails.

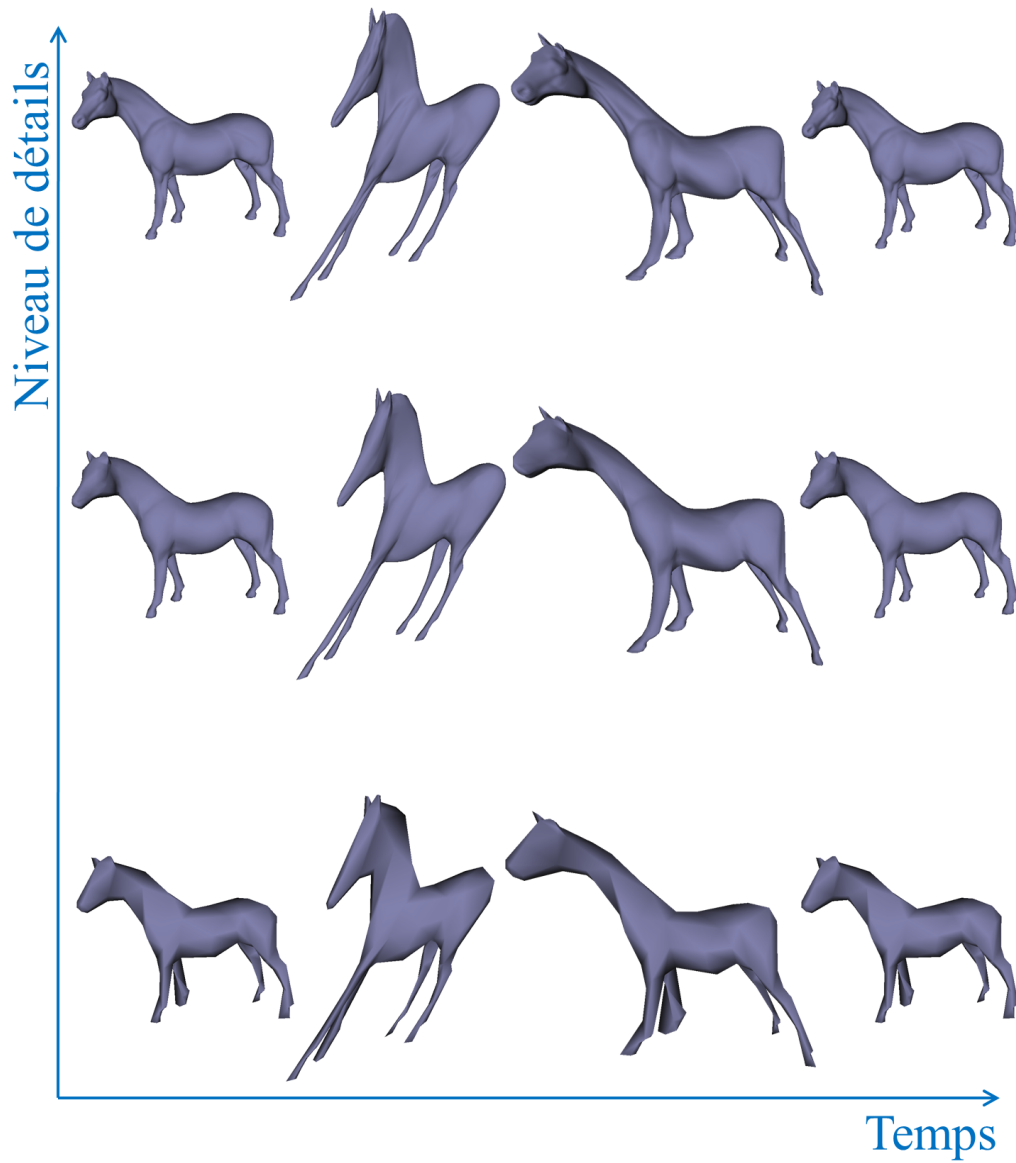


FIGURE 5.10 – En haut, la séquence FLYING HORSE remaillée. Au milieu, la séquence FLYING HORSE remaillée et représentée avec trois niveaux de détails. En bas, la séquence FLYING HORSE remaillée et représentée avec un seul niveau de détails.

TABLE 5.2 – Temps de calcul nécessaire pour remailler la séquence DOLPHIN, mesures étant effectuées sur un processeur 1,66 GHz et 1 Go de RAM.

Opération	Temps
Classification	02sec
Recalage sur le maillage de référence et calcul du maillage moyen	01min 35sec
Simplification (240 triangles) du maillage moyen et paramétrisation	16sec
Remaillage (61440 triangles)	04min 06sec
Temps total	05min 58sec

Chapitre 6

Compression des séquences de maillages semi-réguliers

6.1 Introduction

Comme cela est expliqué dans le chapitre 4, beaucoup de travaux portent sur la compression de séquences de maillages surfaciques à connectivité constante. Cette connectivité constante au cours du temps implique une régularité temporelle qui peut être exploitée lors de la compression de ces séquences, en utilisant par exemple une transformée en ondelettes temporelle [95, 114]. L'échantillonnage spatiale est, par contre, généralement irrégulier, ce qui rend difficile l'utilisation de transformées en ondelettes spatiales.

Une solution face à ce problème d'irrégularité de l'échantillonnage spatiale est d'utiliser une technique de remaillage semi-régulier, comme celle que nous avons présentée dans le chapitre précédent. Cette technique de remaillage nous permet d'obtenir des séquences de maillages semi-réguliers et par la suite nous facilite l'exploitation de la corrélation spatiale.

Nous proposons donc dans ce chapitre un schéma de compression pour les séquences de maillages semi-réguliers qui comprend un filtrage spatio-temporel basé-ondelettes pour exploiter la corrélation dans le temps et dans l'espace. Afin d'optimiser la quantification des coefficients ainsi obtenus, le schéma de compression proposé comprend également une allocation débit basé modèles.

Ce chapitre est organisé de la manière suivante : La section 6.2 présente notre approche pour la compression des séquences de maillages semi-réguliers. La section 6.3 détaille notre analyse spatio-temporelle. La section 6.4 détaille notre allocation débit basée modèles et optimisée pour notre schéma d'analyse. Nous montrons des résultats expérimentaux dans la section 6.5. Nous concluons enfin dans la section 6.6.

6.2 Présentation de l'approche proposée

Nous considérons ici une \mathcal{S}_M , définie par une séquence de T maillages semi-réguliers ayant la même connectivité : $\mathcal{S}_M = \{M_1, M_2, \dots, M_T\}$. La structure du schéma de compression proposé est illustrée figure 6.1.

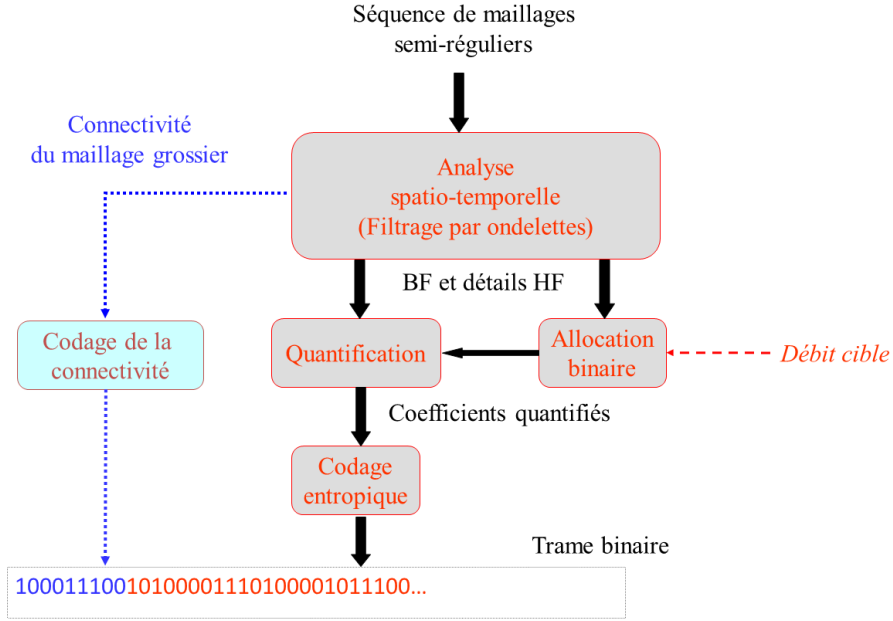


FIGURE 6.1 – Schéma de compression proposé.

Les principales étapes de ce schéma sont les suivantes :

- **analyse spatio-temporelle** : la séquence \mathcal{S}_M a un échantillonnage régulier, à la fois dans le temps et dans l'espace. Par suite, un filtrage temporel basé ondelettes est appliqué, suivi par la transformée spatiale en ondelettes basée Butterfly [79]. Ainsi, nous obtenons une décomposition spatio-temporelle de l'animation en entrée (voir figure 6.3) qui consiste en plusieurs ensembles de détails haute fréquence (coefficients d'ondelettes) temporels et spatiaux, et une séquence de quelques maillages grossiers (signal basse fréquence). La liste de triangles d'un maillage grossier représente toute l'information de connectivité, qui est à coder par la suite.
- **allocation binaire** : ce processus optimise le compromis débit-distorsion, qui est relatif à la quantification des données. L'objectif est de distribuer le débit d'une manière efficace à travers toutes les sous-bandes de coefficients d'ondelettes, en fonction de leur influence sur la qualité de la séquence de maillages reconstruite. Dans cette perspective, ce processus va calculer les pas de quantification optimaux pour chaque sous-bande, pour un débit donné par l'utilisateur.

- **codage entropique** : une fois quantifiées, les sous-bandes de coefficients d'ondelettes sont codées avec le codeur arithmétique contextuel défini dans [114].
- **codage de la connectivité** : pour reconstruire les données codées après stockage et/ou transmission, l'information de connectivité doit être préservée. Pour cela, la liste de triangles formant cette information est codée sans perte (avec la technique de Touma et Gotsman [133]), puis incluse dans la trame binaire.

6.3 Décomposition spatio-temporelle

Afin de réaliser une décomposition spatio-temporelle des séquences de maillages semi-régulières, nous avons choisi d'appliquer successivement une transformée en ondelettes temporelle et une autre spatiale. Ces deux transformées sont implémentées en schéma *lifting* [129].

6.3.1 Transformée en ondelette temporelle

Le principe de la transformée en ondelettes temporelle proposée est de considérer la trajectoire de chaque sommet de l'animation comme étant un signal mono-dimensionnel (voir figure 6.2). Ainsi, une transformée en schéma *lifting* 1D classique est appliquée successivement sur les positions de chaque sommet (n'importe quelle transformée peut être utilisée). A la fin de cette étape, nous obtenons les deux sous-bandes suivantes :

- une sous-bande haute fréquence (HF) H_1^t , qui est définie par $T/2$ trames contenant des détails (coefficients d'ondelettes temporels), soit : $H_1^t = \{h_{1,0}^t, h_{1,1}^t, \dots, h_{1,T/2}^t\}$;
- une sous-bande basse fréquence (BF) L_1^t , qui est définie par $T/2$ trames en approximant la séquence originale (signal basse fréquence), soit : $L_1^t = \{l_{1,0}^t, l_{1,1}^t, \dots, l_{1,T/2}^t\}$.

Nous pouvons obtenir une décomposition temporelle multirésolution en ré-appliquant cette transformée sur le signal basse fréquence obtenu. Cette décomposition correspond à N_t sous-bandes haute fréquences $H_{r^t}^t$ (où r^t représente la résolution temporelle) et au signal basse fréquence $L_{N_t}^t$.

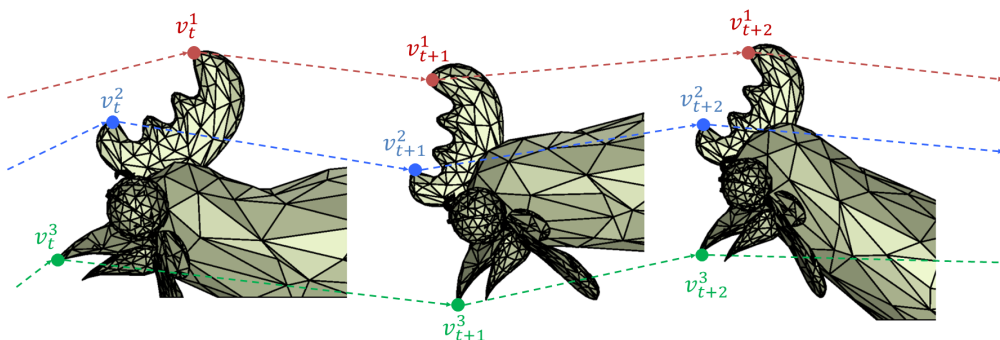


FIGURE 6.2 – Le mouvement de chaque sommet du maillage est considéré comme un signal 1D.

6.3.2 Transformée en ondelettes spatiale

Après l'utilisation de la transformée en ondelette temporelle, nous appliquons une transformée en ondelettes spatiale basée sur le schéma *lifting* basé Butterfly [79], et ce, en considérant toutes les trames de la séquence basse fréquence $L_{N_t}^t$. Pour chaque trame $l_{N_t,k}^t$ de cette séquence, cette étape donne N_s sous-bandes de coefficients spatiaux $h^s l_{r^s,k}^t$ (où r^s représente le niveau de résolution spatiale) et une version grossière $l_{N_s,k}^t$ de la trame en question.

Le principe de cette décomposition est illustré figure 6.3. Dans cette figure, nous avons appliqué 2 niveaux de décomposition temporelle et 2 niveaux de décomposition spatiale sur la séquence DOLPHIN.

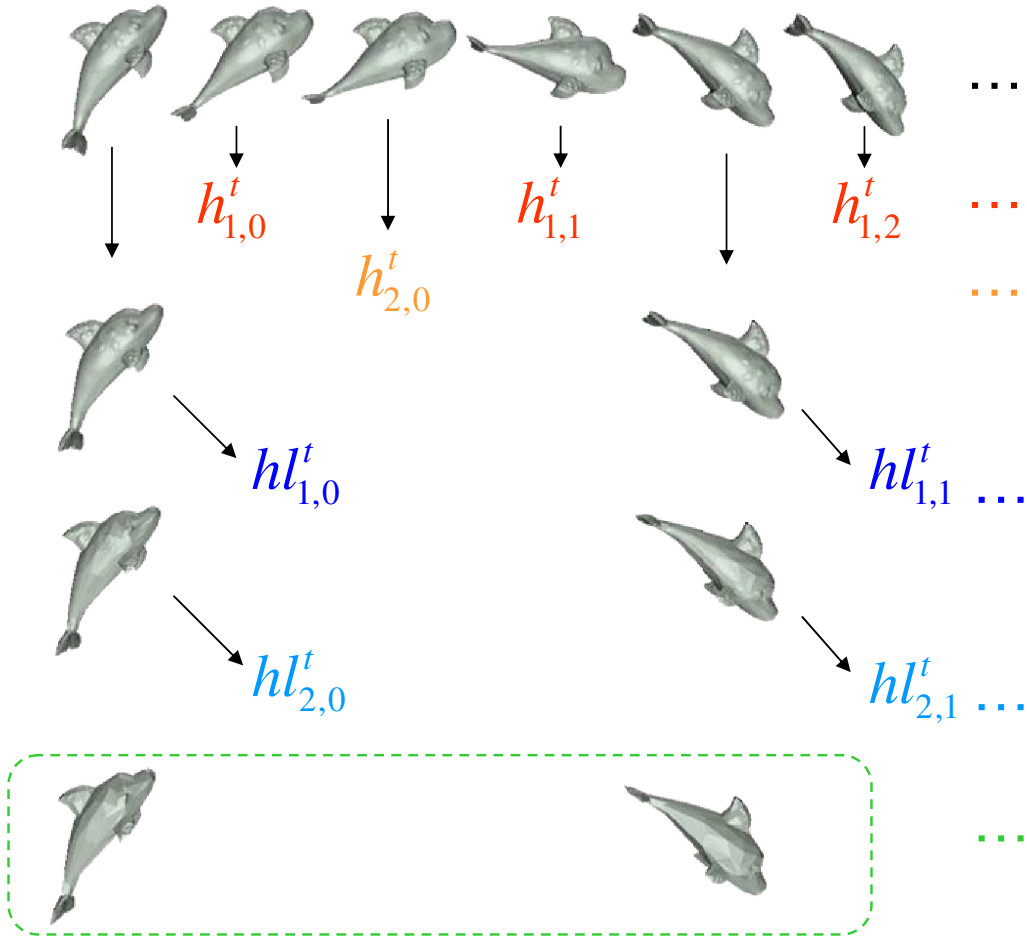


FIGURE 6.3 – Décomposition spatio-temporelle basée sur deux transformées en ondelettes : les coefficients en ondelettes temporels (en rouge et orange) sont obtenus avec 2 niveaux de décomposition temporelle, les coefficients en ondelettes spatiaux (en bleu) sont obtenus avec 2 niveaux de décomposition spatiale. Les maillages dans le rectangle en pointillé (en vert) représente la séquence grossière (le signal basse fréquence spatio-temporel).

Compte tenu de la structure multirésolution obtenue à la fin de cette phase d'analyse, nous allons maintenant aborder le processus d'allocation binaire, qui a pour but d'optimiser la quantification des différentes sous-bandes.

6.4 Allocation binaire

Avec une structure multirésolution, il n'est pas possible d'utiliser le même pas de quantification pour toutes les sous-bandes, étant donné que l'énergie du signal de départ n'est pas uniformément répartie. Rappelons qu'une fois que les transformées en ondelettes appliquées, la plus grande partie de l'information initiale est concentrée dans les sous-bandes de basses résolutions, tandis que les sous-bandes de hautes résolutions contiennent généralement des détails négligeables. Ainsi, nous utilisons généralement un processus d'allocation binaire pour calculer les pas de quantification optimaux pour chaque sous-bande, afin d'obtenir le meilleur compromis entre débit (taille de la trame binaire) et distorsion (qualité des données reconstruites) [111].

Dans notre cas, le but de l'allocation binaire est de trouver les pas de quantification optimaux $\{q_{i,j}^s, q_{i,j}^t\}$, où $q_{i,j}^s$ et $q_{i,j}^t$ sont les pas de quantification des sous-bandes spatiales et temporelles, respectivement. Ces pas de quantifications optimaux minimisent l'erreur de reconstruction (ou distorsion totale) $D_T(\{q_{i,j}^s, q_{i,j}^t\})$ pour un débit cible donné par l'utilisateur R_{target} . Ceci peut être formulé par le problème suivant :

$$(\mathcal{P}) \begin{cases} \text{minimiser} & D_T(\{q_{i,j}^s, q_{i,j}^t\}) \\ \text{sous la contrainte} & R_T(\{q_{i,j}^s, q_{i,j}^t\}) = R_{target}. \end{cases} \quad (6.1)$$

Pour pouvoir résoudre ce problème, nous devons tout d'abord formuler la distorsion totale D_T .

6.4.1 Formulation de la distorsion totale

Dans ce qui suit, nous détaillons comment nous avons formulé théoriquement la distorsion totale D_T . Pour cela, nous commençons par détailler le cas avec uniquement une décomposition temporelle [134]. Par la suite, nous détaillons le cas avec uniquement une décomposition spatiale [111]. Enfin, nous combinons ces deux cas pour traiter la décomposition spatio-temporelle proposée.

Décomposition temporelle

Nous considérons un schéma de codage basé sur un seul niveau de décomposition temporelle (voir figure 6.4).

Dans la figure 6.4, la distorsion totale D_T peut être formulée comme étant la somme pondérée des erreurs quadratiques moyennes $(\sigma_{01}^t)^2$ et $(\sigma_{11}^t)^2$, qui sont relatives à la quantification du signal basse fréquence et les coefficients d'ondelettes, respectivement [134]. Nous obtenons alors :

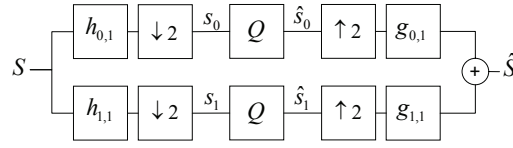


FIGURE 6.4 – Modèle d'un schéma de codage basé sur un seul niveau de décomposition temporelle, où $(h_{0,1}, g_{0,1})$ et $(h_{1,1}, g_{1,1})$ sont les filtres d'analyse/synthèse passe-bas et passe-haut, respectivement. Q représente le quantificateur, et $\downarrow 2$ et $\uparrow 2$ sont les étapes de sous-échantillonnage et de sur-échantillonnage, respectivement.

$$D_T = \frac{1}{2} w_{01}^t (\sigma_{01}^t)^2 + \frac{1}{2} w_{11}^t (\sigma_{11}^t)^2, \quad (6.2)$$

où w_{ij}^t sont les poids dus à la non-orthogonalité des filtres d'ondelettes temporels ($i = 0$, pour la sous-bande basse fréquence, $i = 1$, pour la sous-bande haute fréquence, et j représente le niveau de résolution). Ces poids sont donnés par :

$$w_{01}^t = \frac{2}{N} \text{tr}((G_{01}^t)^T G_{01}^t), \quad (6.3)$$

et

$$w_{11}^t = \frac{2}{N} \text{tr}((G_{11}^t)^T G_{11}^t), \quad (6.4)$$

où N est le nombre de sommets et G_{ij}^t est la matrice de reconstruction relative au filtre symétrique g_{ij}^t .

Nous considérons maintenant un schéma de codage basé sur N_t niveaux de décomposition temporelle. En utilisant l'expression (6.2), nous pouvons formuler l'erreur de reconstruction à chaque niveau de décomposition $(n - 1)$, ($2 \leq n \leq N_t$), à partir de l'expression suivante :

$$(\sigma_{0(n-1)}^t)^2 = \frac{1}{2} w_{0n}^t (\sigma_{0n}^t)^2 + \frac{1}{2} w_{1n}^t (\sigma_{1n}^t)^2. \quad (6.5)$$

Compte-tenu du fait que nous utilisons les mêmes filtres de synthèse à chaque niveau de décomposition, nous avons :

$$g_{0i} = g_{01}, \quad \forall i \in [2, N_t], \quad (6.6)$$

et

$$g_{1i} = g_{11} \quad \forall i \in [2, N_t], \quad (6.7)$$

Par conséquent les poids sont similaires, quelque soit le niveau :

$$w_{0i}^t = w_{01}^t = w_{lf}^t \quad \forall i \in [2, N_t], \quad (6.8)$$

et

$$w_{1i}^t = w_{11}^t = w_{hf}^t \quad \forall i \in [2, N_t]. \quad (6.9)$$

En combinant les nouveaux poids des expressions (6.2), (6.5), (6.8) et (6.9) nous obtenons finalement :

$$D_T = \frac{1}{2^{N_t}} (w_{lf}^t)^{N_t} (\sigma_{0N_t}^t)^2 + \sum_{j=1}^{j=N_t} \left(\frac{1}{2^j} (w_{lf}^t)^{j-1} w_{hf}^t (\sigma_{1j}^t)^2 \right). \quad (6.10)$$

Décomposition spatiale

Par analogie avec ce que nous venons de détailler pour la décomposition temporelle, la distorsion pour N_s niveaux de décomposition spatiale d'une séquence de maillages triangulaires peut être estimée en utilisant l'expression suivante :

$$(\sigma_s)^2 = (w_{lf}^s)^{N_s} (\sigma_{0N_s}^s)^2 + \sum_{j=1}^{j=N_s} \left((w_{lf}^s)^{j-1} w_{hf}^s (\sigma_{1j}^s)^2 \right), \quad (6.11)$$

où w_{lf}^s et w_{hf}^s sont les poids dus à la non-orthogonalité des filtres d'ondelettes spatiaux pour les sous-bandes basse fréquence et haute fréquence, respectivement.

Décomposition spatio-temporelle

Nous pouvons maintenant généraliser le critère de distorsion pour une décomposition spatio-temporelle avec N_t niveaux de décomposition temporelle et N_s niveaux de décomposition spatiale. En effet, la décomposition spatiale s'effectue sur la sous-bande basse fréquence obtenue après la décomposition temporelle. Dans l'expression (6.10) qui formule la distorsion totale dans le cas d'une décomposition temporelle uniquement, la distorsion de la sous-bande basse fréquence est $(\sigma_{0N_t}^t)^2$. Il suffit de remplacer cette distorsion par la formulation de la distorsion obtenue dans le cas d'une décomposition spatiale uniquement qui est donnée par l'expression (6.11). Par la suite, nous obtenons la distorsion totale pour une décomposition spatio-temporelle :

$$D_T = \frac{1}{2^{N_t}} (w_{lf}^t)^{N_t} \frac{1}{N_{LF}} \sum_{f=1}^{f=N_{LF}} \left((w_{lf}^s)^{N_s} (\sigma_{0N_s}^s)^2 \sum_{j=1}^{j=N_s} \left((w_{lf}^s)^{j-1} w_{hf}^s (\sigma_{1j}^s)^2 \right) \right) + \sum_{j=1}^{j=N_t} \left(\frac{1}{2^j} (w_{lf}^t)^{j-1} w_{hf}^t (\sigma_{1j}^t)^2 \right) \quad (6.12)$$

Pour chaque trame de la sous-bande basse fréquence temporelle, nous obtenons N_s sous-bandes spatiales. Ainsi, le nombre des sous-bandes spatiales est égal à $N_{LF} N_s$, où N_{LF} est le nombre de trames dans la sous-bande basse fréquence temporelle.

6.4.2 Allocation binaire basée modèles

Comme expliqué dans [112], une allocation binaire est généralement un processus avec un algorithme itératif. A chaque itération, nous avons besoin d'évaluer la distorsion

(et les débits associés) de chaque sous-bande pour un ensemble de pas de quantification afin de vérifier la convergence de l'algorithme. Par suite, nous pouvons effectuer une quantification/dé-quantification réelle et calculer les distorsions réelles, qui sont obtenues pour chaque sous-bande. Cette démarche demande un énorme temps de calcul, surtout quand nous travaillons avec de grandes séquences de maillages. Une méthode efficace pour éviter ce problème et réduire le temps de calcul est d'utiliser des modèles statistiques afin d'estimer les distorsions et débit de chaque sous-bande sans avoir besoin d'effectuer une étape de quantification/dé-quantification préalable [111]. La figure 6.5 montre la distribution des coefficients d'ondelettes spatio-temporels pour la séquence FLYING VENUS. Remarquons que les coefficients d'ondelettes suivent une distribution Gaussienne, et ce, quelque soit le niveau de décomposition. Par conséquent, nous pouvons utiliser l'allocation binaire basée modèles proposée dans [112]. Pour plus d'informations, nous invitons les lecteurs à se reporter à cette référence.

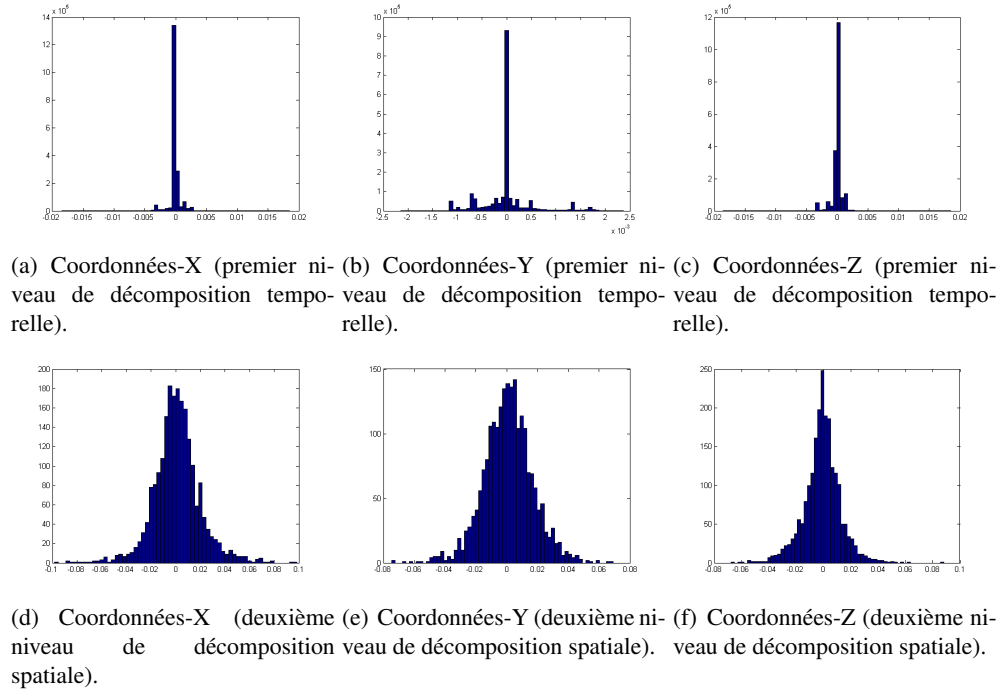


FIGURE 6.5 – Distributions typiques des coefficients d'ondelettes de la séquence de mail-
lages semi-réguliers FLYING VENUS avec la décomposition spatio-temporelle proposée.

6.5 Résultats expérimentaux

Dans cette section, nous allons comparer le schéma de compression proposé avec le codeur basé ondelettes temporelles ([PA07]) proposé par Payan et Antonini en 2007 [114] et appliqué directement sur la séquence irrégulière originale. Afin de montrer l'impact de notre allocation débit optimale, nous nous sommes comparés également avec une

approche basée analyse spatio-temporelle mais sans l'allocation binaire optimale et qui se résume ainsi :

- analyse spatio-temporelle,
- codage des sous-bandes de coefficients temporelles avec le codeur proposé dans [114],
- codage des sous-bandes de coefficients spatiales avec le codeur PGC proposé dans [79].

Dans tous les tests qui vont suivre, nous avons utilisé le filtre (4,2) pour la transformée en ondelettes temporelle.

Les figures 6.6, 6.7, et 6.8 montrent la distorsion moyenne (calculée sur tous les maillages de la séquence en utilisant le même critère de qualité que précédemment en fonction de différents débits, pour la séquence DOLPHIN, FLYING VENUS et FLYING HORSE.

Nous remarquons qu'avec notre schéma de compression (avec ou sans allocation binaire optimale), nous améliorons significativement les taux de compression par rapport au codeur proposé dans [114]. Ces résultats prouvent l'intérêt d'utiliser une analyse spatio-temporelle. En effet, en utilisant des séquences de maillages semi-réguliers, nous pouvons appliquer efficacement une transformée en ondelettes spatiale. Cette transformée spatiale permet d'améliorer les taux de compression en éliminant la redondance spatial.

Nous remarquons également que nous obtenons les meilleurs résultats en utilisant l'allocation binaire optimale. En effet, optimiser la distribution du débit à travers les différentes sous-bandes temporelles et spatio-temporelles avec une allocation binaire optimale permet d'améliorer significativement les taux de compression.

Les figures 6.9, 6.10 et 6.11 montrent l'évolution de la distorsion, maillage par maillage, à des débits donnés. Globalement, nous observons que nous obtenons les meilleurs résultats, à savoir l'erreur la plus faible, ce qui confirme les résultats cités dans le paragraphe précédent.

Nous remarquons qu'en utilisant un débit plus faible, nous obtenons une erreur de reconstruction plus faible pour la majorité des trames. En plus, notre approche tends à régulariser la qualité à travers toutes les trames (élimination de pics de distorsion sur certains maillages). C'est un résultat très intéressant, puisqu'en utilisant notre méthode, nous arrivons à garder une qualité plus régulière le long de la séquence, ce qui permet d'obtenir une animation fluide et homogène.

6.6 Conclusion

Dans ce chapitre, nous avons présenté un nouveau schéma de compression des séquences de maillages semi-réguliers. Nous avons développé un schéma de codage spatio-temporel basé ondelettes pour ces séquences. Ces séquences de maillages semi-réguliers ont l'avantage de faciliter la décomposition en ondelettes grâce à leurs échantillonnage régulier. Nous avons aussi proposé une allocation binaire basée modèles, qui permet de

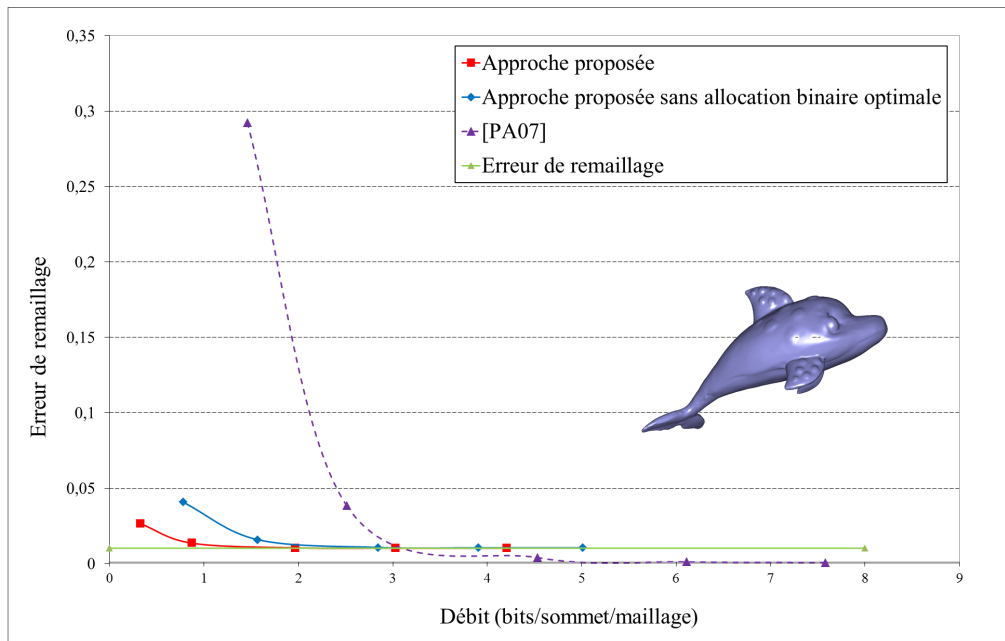


FIGURE 6.6 – Courbes d'évolution débit-distorsion pour la séquence DOLPHIN.

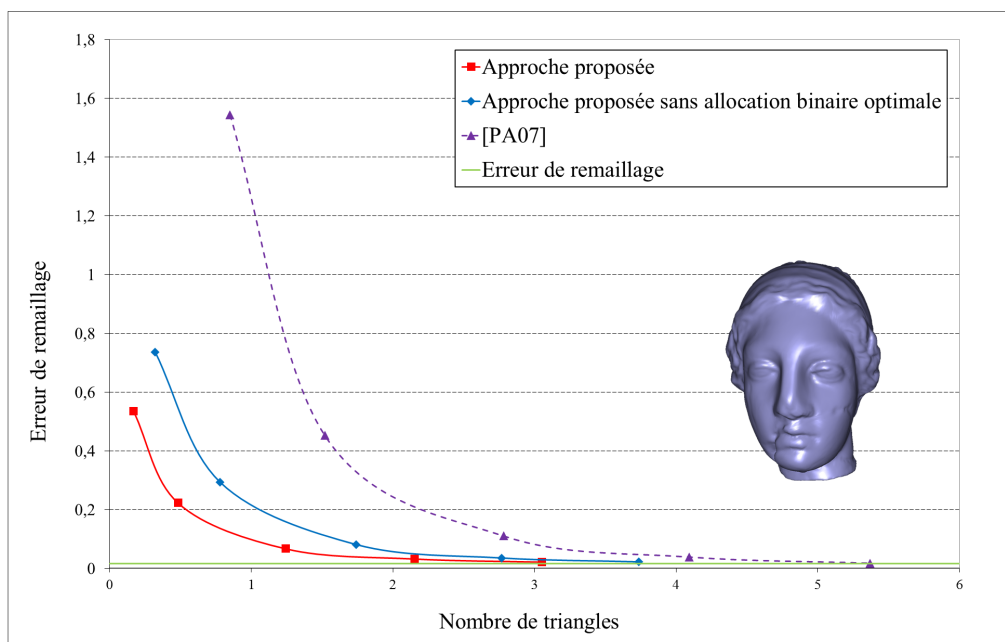


FIGURE 6.7 – Courbes d'évolution débit-distorsion pour la séquence FLYING VENUS.

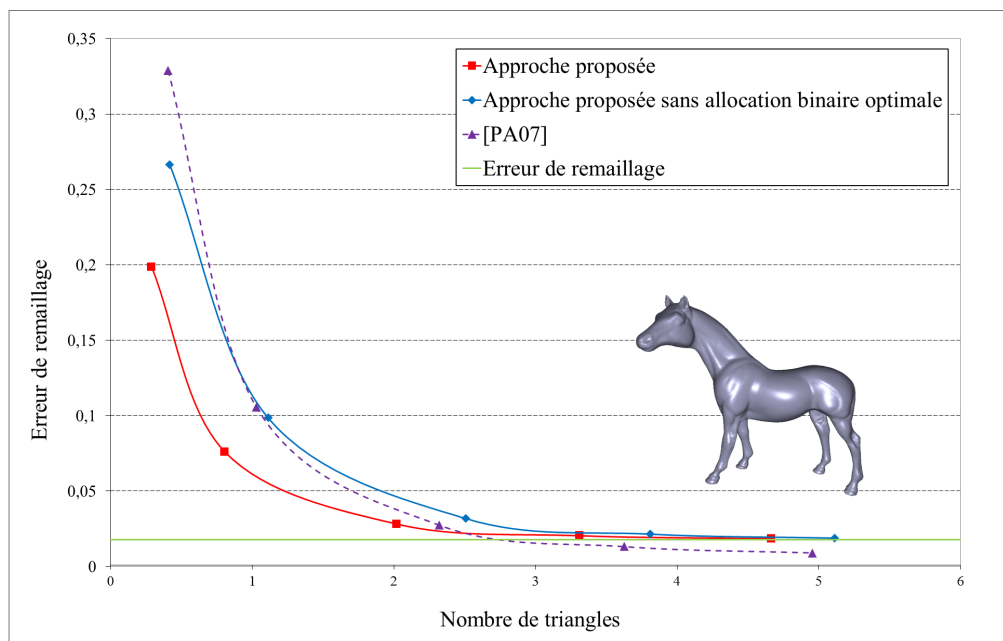


FIGURE 6.8 – Courbes d'évolution débit-distorsion pour la séquence FLYING HORSE.

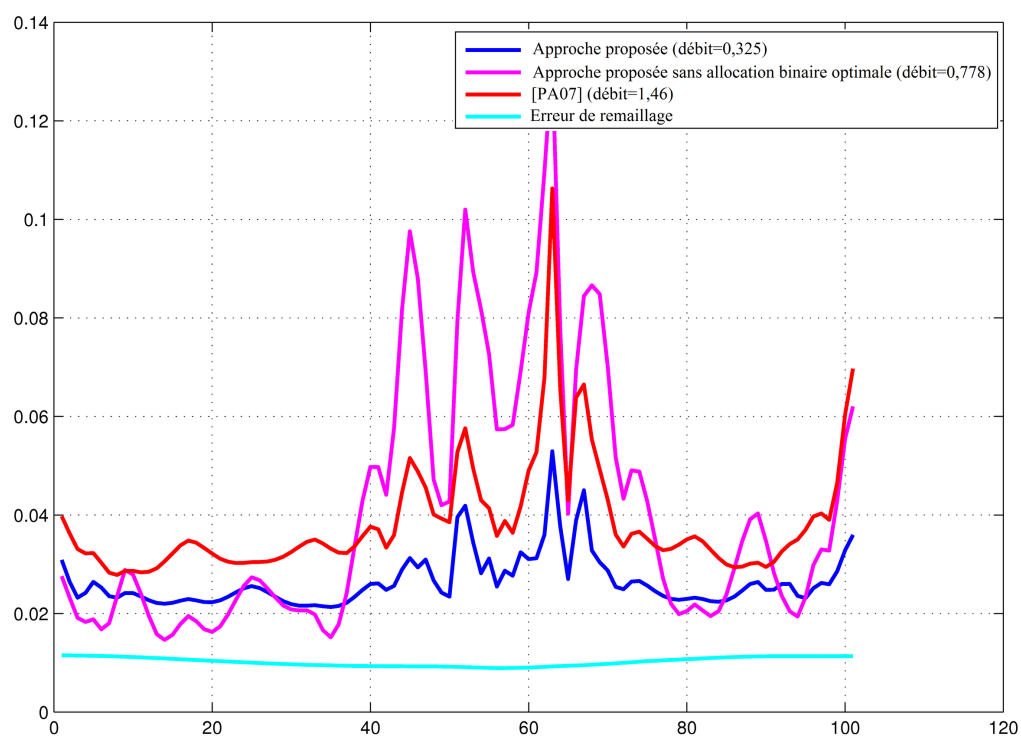


FIGURE 6.9 – Courbes d'évolution distorsion-trame pour la séquence DOLPHIN.

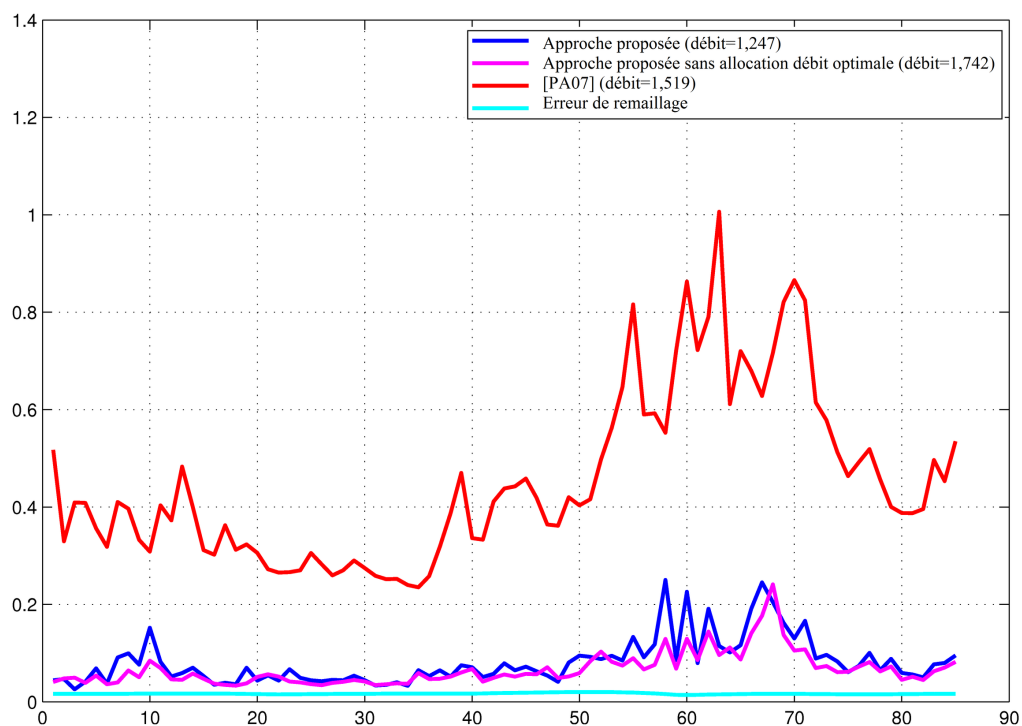


FIGURE 6.10 – Courbes d'évolution distorsion-trame pour la séquence FLYING VENUS.

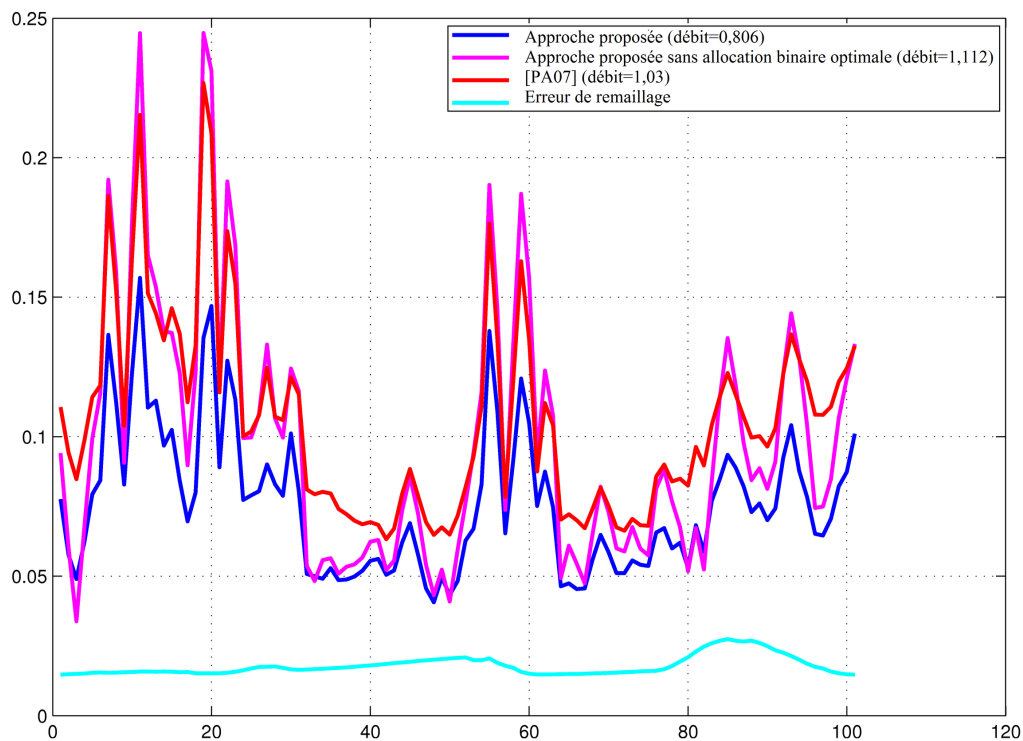


FIGURE 6.11 – Courbes d'évolution distorsion-trame pour la séquence FLYING HORSE.

calculer les pas de quantification optimaux pour toutes les sous-bandes temporelles et spatiales. Les résultats expérimentaux ont montré que l'approche proposée améliore les performances de compression par rapport au codeur décrit dans [114].

Conclusion

Dans cette deuxième partie du manuscrit, nous avons développé des méthodes concernant le remaillage semi-régulier et la compression basée ondelettes pour les animations 3D définies par des séquences de maillages à connectivité constante.

Remaillage semi-régulier dynamique

Notre méthode de remaillage semi-régulier est basée le remailleur statique proposé dans [87]. Nous commençons par recalcr tous les maillages de la séquence sur un maillage choisi arbitrairement. Par la suite, nous calculons un maillage moyen à partir des maillages recalés. Ensuite, nous remaillons ce maillage moyen avec un remailleur statique similaire à celui proposé dans [87]. Enfin, nous déformons le maillage semi-régulier ainsi obtenu afin de créer la séquence semi-régulière.

Nous envisageons de traiter dans le futur les deux thèmes suivants :

- trouver une heuristique afin de choisir le maillage de référence M_{ref} qui permet d’obtenir le meilleur recalage et par la suite la plus faible erreur de remaillage ;
- prendre en compte le mouvement des sommets lors de la simplification afin de mieux préserver les sommets qui bougent le plus dans le temps ;

Compression basée ondelettes et analyse spatio-temporelle

Notre schéma de compression pour les séquences de maillages semi-réguliers est basée sur une analyse spatio-temporelle et une allocation débit optimale. Nous commençons par appliquer une transformée en ondelettes temporelle sur la séquence. Par la suite, nous appliquons une transformée en ondelettes spatiale sur la séquence basse fréquence temporelle. Ensuite, nous codons la basse fréquence spatio-temporelle et les différents coefficients en ondelettes ainsi obtenus. Pour optimiser le codage, nous utilisons une allocation binaire optimale basée modèles. Cette allocation binaire prend en compte l’impact des erreurs de quantification de chaque sous-bande sur l’erreur de reconstruction finale.

Nous envisageons de traiter dans le futur les deux thèmes suivants :

- inclure une compensation en mouvement dans notre schéma de compression ;
- construire une transformée en ondelettes 3D spatio-temporelle.

Conclusion générale

Les objectifs visés dans cette thèse sont divisés en deux catégories :

- proposer des approches pour le remaillage semi-réguliers des maillages et des séquences de maillages surfaciques ;
- améliorer les schémas de compression existants pour les maillages semi-réguliers et les séquences de maillages en améliorant principalement l'étape d'analyse basée sur les ondelettes fréquemment utilisées dans le domaine.

Nous avons atteint nos objectifs en proposant une nouvelle approche pour le remaillage semi-régulier des maillages surfaciques. Notre approche exploite les diagrammes de Voronoï et la relaxation de Lloyd pour produire des maillages semi-réguliers. En effet, nous construisons un diagramme de Voronoï multirésolution sur le maillage irrégulier en entrée et nous utilisons la relaxation de Lloyd pour le remplacement des sommets durant les étapes de simplification et de subdivision. Notre approche nous a permis d'obtenir des maillages semi-réguliers de bonne qualité et nous a permis également de créer facilement des maillages semi-réguliers adaptatifs.

Nous avons également proposé une technique de remaillage semi-régulier pour les séquences de maillages surfaciques. Notre approche basée sur les travaux de [87] propose de remailler un maillage moyen obtenu à partir des différents maillages de la séquence recalés sur un maillage de référence. Cette technique nous permet également d'obtenir des séquences de maillages de bonnes qualités.

Pour améliorer les schémas de compression existants, nous avons proposé d'optimiser deux transformées en ondelettes pour les maillages semi-réguliers. Notre optimisation consiste à calculer les filtres de prédiction afin de maximiser la parcimonie des coefficients d'ondelettes et par la suite d'adapter les filtres de mise à jour afin de prendre en compte les modifications des paramètres de prédiction et de satisfaire certaines propriétés de la transformée.

Nous avons également proposé un schéma de compression basé sur une analyse spatio-temporelle et une allocation binaire optimale pour les séquences de maillages semi-réguliers. En effet, nous appliquons deux transformées en ondelettes (une temporelle et l'autre spatiale) sur la séquence en entrée. Ceci, nous permet d'obtenir une sous-bande basse résolution spatio-temporelle et des sous-bandes de coefficients d'on-

delettes, que nous codons par la suite, en optimisant les pas de quantification de chaque sous-bande à l'aide d'une allocation binaire optimale et basée modèles.

Perspectives

Nos travaux de recherches présentent plusieurs perspectives, qui permettent d'améliorer les performances de nos approches pour le remaillage semi-réguliers des maillages surfaciques et des séquences de maillages surfaciques et les performances de nos schémas de compression pour les maillages semi-réguliers et séquences de maillages semi-réguliers.

Remaillage semi-régulier des maillages surfaciques et des séquences de maillages surfaciques

Nous envisageons d'améliorer notre approche basée sur la relaxation de Lloyd pour le remaillage semi-régulier en ajoutant une gestion des bords et des arêtes saillantes. En effet, nous voulons dans le futur, améliorer l'initialisation et le repositionnement des sites de Voronoï afin de mieux préserver l'allure générale du maillage traité.

Nous envisageons également, d'étendre notre approche basée sur la relaxation de Lloyd aux séquences de maillages surfaciques. Pour ceci, nous envisageons de construire un diagramme de Voronoï multirésolution sur la séquence en entrée tout en prenant en compte les mouvements des sommets. Toujours dans l'optique d'améliorer notre approche pour le remaillage dynamique, nous voulons dans le futur, trouver une heuristique qui nous permet de choisir le maillage de référence M_{ref} afin d'obtenir le meilleur recalage possible et par conséquent une plus faible erreur de remaillage.

Compression des maillages semi-réguliers et séquences de maillages semi-réguliers

Nous envisageons de modifier le codeur géométrique proposé par Payan et Antonini [112] afin de le rendre adaptatif et ainsi pouvoir tirer profit de notre méthode adaptative de remaillage semi-régulier. Nous envisageons également de modifier ce même codeur, afin de le rendre basé-régions. En effet, il serait judicieux de segmenter le maillage semi-régulier et optimiser la transformée en ondelettes pour chaque région séparément, ainsi nous pourrions améliorer les taux de compression.

Nous envisageons également d'inclure une compensation en mouvement dans notre schéma de compression pour les séquences de maillages. Et par la suite d'étendre notre technique d'optimisation des transformées en ondelettes pour les maillages semi-réguliers à notre schéma d'analyse spatio-temporelle pour les séquences de maillages semi-réguliers, et ce dans le but d'améliorer les taux de compression des séquences de maillages en maximisant la parcimonie des coefficients d'ondelettes obtenus.

Bibliographie

- [1] <http://en.wikipedia.org/wiki/lloyd> [citée p. 45]
- [2] <http://www.disney.fr/toy-story>. [citée p. 3, 161]
- [3] <http://www.mycrYSIS.com/>. [citée p. 4, 161]
- [4] <http://www.nibib.nih.gov>. [citée p. 4, 161]
- [5] J.-H. AHN, C.-S. KIM, C.-C. Jay KUO et Y.-S. Ho : Motion-compensated compression of 3d animation models. *Electronics Letters*, 37(24):1445–1446, 2001. [citée p. 104, 110]
- [6] M. AHN, I. GUSKOV et S. LEE : Out-of-core remeshing of large polygonal meshes. *IEEE Transactions on Visualization and Computer Graphics*, 12:1221–1228, Septembre 2006. [citée p. 31, 39, 64, 115]
- [7] M. ALEXA et W. MULLER : Representing animations by principal components. *Computer Graphics Forum*, 3, 2000. [citée p. 104, 111, 112]
- [8] P. ALLIEZ, G. UCELLI, C. GOTSMAN et M. ATTENE : Recent advances in remeshing of surfaces. In L. FLORIANI, M. SPAGNUOLO, G. FARIN, H.-C. HEGE, D. HOFFMAN, C.R. JOHNSON et K. POLTHIER, éditeurs : *Shape Analysis and Structuring*, Mathematics and Visualization, pages 53–82. Springer Berlin Heidelberg, 2008. [citée p. 35, 36]
- [9] R. AMJOUN, R. SONDRSHAUS et W. STRASSER : Compression of complex animated meshes. In Tomoyuki NISHITA, Qunsheng PENG et Hans-Peter SEIDEL, éditeurs : *Advances in Computer Graphics*, volume 4035 de *Lecture Notes in Computer Science*, pages 606–613. Springer Berlin / Heidelberg, 2006. [citée p. 104, 110]
- [10] R. AMJOUN et W. STRASSER : Predictive-dct coding for 3d mesh sequences compression. *Journal of Virtual Reality and Broadcasting*, 5(6), Juillet 2008. [citée p. 104, 110]
- [11] N. ASPERT, D. SANTA-CRUZ et T. EBRAHIMI : Mesh : Measuring errors between surfaces using the hausdorff distance. In *IEEE International Conference in Multimedia and Expo (ICME)*, Lausanne, Suisse, Août 2002. [citée p. 32]
- [12] F. AURENHAMMER et R. KLEIN : *Voronoi Diagrams*. [citée p. 42]
- [13] B.G. BAUMGART : Winged edge polyhedron representation. Rapport technique, Stanford University, Stanford, Etats Unis, 1972. [citée p. 26]

- [14] A. BENAZZA-BENYAHIA, J.-C. PESQUET, J. HATTAY et H. MASMOUDI-KHODJA : Block-based adaptive vector lifting schemes for multispectral image coding. *EURASIP International Journal of Image and Video Processing*, Janvier 2007. [citée p. 66]
- [15] M. BERTRAM : Biorthogonal loop-subdivision wavelets. *Computing*, 72:29–39, 2004. [citée p. 65, 77, 78, 79, 80, 82, 84, 92, 97, 98, 162]
- [16] J. BLOOMENTHAL : An implicit surface polygonizer. In *Graphics Gems IV*, pages 324–349. Academic Press, 1994. [citée p. 18]
- [17] J.-D. BOISSONNAT et F. CAZALS : Smooth surface reconstruction via natural neighbour interpolation of distance functions. In *Proceedings of the Sixteenth Annual Symposium on Computational Geometry*, pages 223–232, 2000. [citée p. 17]
- [18] G.P. BONNEAU, S. HAHMANN et G.M. NIELSON : Blac-wavelets : a multiresolution analysis with non-nested spaces. In *Visualization Proceedings*, pages 43–48. IEEE, 1996. [citée p. 63]
- [19] M. BOTSCH : *High Quality Surface Generation and Efficient Multiresolution Editing Based on Triangle Meshes*. Thèse de doctorat, RWTH Aachen, 2005. [citée p. 23, 161]
- [20] M. BOTSCH, L. KOBELT, M. PAULY, P. ALLIEZ et B. LÉVY : *Polygon Mesh Processing*. A K Peters, 2009. [citée p. 24]
- [21] Y. BOULFANI : *Compression au fil de l'eau de séquences de maillages 3D par analyse multirésolution*. Thèse de doctorat, Université de Nice-Sophia Antipolis, 2010. [citée p. 104, 110, 114]
- [22] Y. BOULFANI-CUISINAUD, M. ANTONINI et F. PAYAN : Motion-base mesh clustering for medwt compression of 3d animated meshes. In *Proceedings of EUSIPCO*, Pologne, Septembre 2007. [citée p. 117]
- [23] H.M. BRICENO, P.V. SANDER, L. McMILLAN, S. GOTLER et H. HOPPE : Geometry videos : a new representation for 3d animations. In *ACM Symposium Computer Animation*, pages 136–146, 2003. [citée p. 104, 111, 112]
- [24] P.J.C. BROWN et C.T. FAIGLE : A robust efficient algorithm for point location in triangulations. Technical report, Cambridge University, Février 1997. [citée p. 122]
- [25] E. CATMULL et J. CLARK : Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10:350–355, 1978. [citée p. 21, 23, 64]
- [26] J.-W. CHO, H.-K. KIM, S. VALETTE, H.-Y. JUNG et R. PROST : A compression method for 3-d mesh sequences combining wavelet-based multi-resolution analysis and multi-order differential coding. In *IPCV*, pages 1008–1011, 2009. [citée p. 104, 114]
- [27] J.W. CHO, M.S. KIM, S. VALETTE, H.Y. JUNG et R. PROST : 3d dynamic mesh compression using wavelet-based multiresolution analysis. In *IEEE International Conference on Image Processing (ICIP'06)*, Octobre 2006. [citée p. 104, 113, 114]
- [28] J.W. CHO, M.S. KIM, S. VALETTE, H.Y. JUNG et R. PROST : A 3d meshes sequence coding using the combination of spatial and temporal wavelet analysis. In *International Conference on Computer Vision, Computer Graphics Collaboration Techniques and Application*, pages 389–399, Rocquencourt, France, Mars 2007. [citée p. 104, 113]

- [29] A. CHOUROU, M. ANTONINI et A. BENAZZA-BENYAHIA : 3d mesh coding through region based segmentation. *In ICASSP*, pages 1381–1384. IEEE, 2008. [citée p. 64, 66]
- [30] M.M. CHOW : Optimized geometry compression for real-time rendering. *In Proceedings of the 8th Conference on Visualization, VIS*, Los Alamitos, Etats Unis, 1997. IEEE Computer Society Press. [citée p. 31]
- [31] G. CIRIO, G. LAVOUÉ et F. DUPONT : A framework for data-driven progressive mesh compression. *In International Conference on Computer Graphics Theory and Applications*, France, 2010. [citée p. 64, 66, 104, 115]
- [32] S. DAHLKE, W. DAHMEN, I. WEINREICH et E. SCHMITT : Multiresolution analysis and wavelets on s_2 and s_3 . *Numerical Functional Analysis and Optimization*, 16(1,2):19–41, 1995. [citée p. 65]
- [33] I. DAUBECHIES : Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41:909–996, 1988. [citée p. 63]
- [34] B. DELAUNAY : Sur la sphère vide. a la mémoire de georges voronoi. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskikh i Estestvennyh Nauk*, 7:93–800, 1934. [citée p. 42]
- [35] L. DENIS, S. M. SATTI, A. MUNTEANU, J. CORNELIS et P. SCHELKENS : Scalable intraband and composite wavelet-based coding of semiregular meshes. *IEEE Transactions on Multimedia*, 12:773–789, Juillet 2010. [citée p. 64, 66, 104, 115]
- [36] R. DESCARTES : *Principia Philosophiae*, Amsterdam. Ludovicus Elzevirius, 1644. [citée p. 40]
- [37] T. DEY, H. EDELSBRUNNER, S. GUHA et D. NEKHAYEV : Topology preserving edge contraction, 1998. [citée p. 120]
- [38] G.L. DIRICHLET : Über die reduktion der positiven quadratischen formen mit drei unbestimmten ganzen zahlen. *J. Reine Angew. Math*, 40:209–227, 1850. [citée p. 41]
- [39] D. DONOHO et Y. TSAIG : Fast solution of l_1 -norm minimization problems when the solution may be sparse. *IEEE Transactions on Information Theory*, 54:4789–4812, 2008. [citée p. 71]
- [40] D.L. DONOHO : Interpolating wavelet transforms, 1992. [citée p. 65]
- [41] D. DOO et M. SABIN : Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10:356–360, 1978. [citée p. 21, 23]
- [42] Q. DU, V. FABER et M. GUNZBURGER : Voronoi tessellations : Applications and algorithms. *SIAM Review*, 41:637–676, Décembre 1999. [citée p. 42, 43]
- [43] T. DUCHAMP, A. CERTAIN, T. DEROSE et W. STUETZLE : Hierarchical computation of pl harmonic embeddings, 1997. [citée p. 38, 122]
- [44] J.-L. DUGELAY, A. BASKURT et M. DAOUDI : *3D Object Processing : Compression, Indexing and Watermarking*. John Wiley & Sons, 2008. [citée p. 16, 17, 18, 23, 164]
- [45] N. DYN, D. LEVIN et J.A. GREGORY : A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9:160–169, Avril 1990. [citée p. 53, 64]

- [46] M. ECK, T. DeROSE, T. DUCHAMP, H. HOPPE, M. LOUNSBERRY et W. STUETZLE : Multiresolution analysis of arbitrary meshes. *In Proceedings of the 22th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 173–182, New York, Etats Unis, 1995. ACM. [citée p. 31, 38, 64, 115]
- [47] G. COLLINS et A. HILTON : Rigid transform basis for animation compression and level of detail. *In Vision Video and Graphics*, pages 21–28, Royaume Uni, 2005. [citée p. 104, 110]
- [48] F. PEREIRA et T. EBRAHIMI : *The mpeg-4 book*. Springer-Verlag,, 2003. [citée p. 111]
- [49] G. FARIN : *Curves and Surfaces for CAD : A Practical Guide*. Academic Press, 1993. [citée p. 21]
- [50] M.S. FLOATER : Parametrization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, 14:231–250, 1997. [citée p. 38]
- [51] M.S. FLOATER et K. HORMANN : Surface parameterization : a tutorial and survey. *In N.A. DODGSON, M.S. FLOATER et M.A. SABIN, éditeurs : Advances in Multiresolution for Geometric Modelling*, pages 157–186. Springer Verlag, 2005. [citée p. 37]
- [52] W. FREEDEN et U. WINDHEUSER : Spherical wavelet transform and its discretization. *Advances in Computational Mathematics*, 5:51–94, 1996. [citée p. 65]
- [53] I. FRIEDEL, P. SCHRÖDER et A. KHODAKOVSKY : Variational normal meshes. *ACM Transactions on Graphics*, 23:1061–1073, 2004. [citée p. 31, 39, 64, 115]
- [54] M. GARLAND : *Quadric-Based Polygonal Surface Simplification*. Thèse de doctorat, Carnegie Mellon University, 1998. [citée p. 120]
- [55] M. GARLAND et P.S. HECKBERT : Surface simplification using quadric error metrics. *In SIGGRAPH : Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pages 209–216, New York, Etats Unis, 1997. ACM Press/Addison-Wesley Publishing Co. [citée p. 119]
- [56] P. GIOIA : Reducing the number of wavelet coefficients by geometric partitioning. *Computational Geometry Theory and Applications*, 14:25–48, Novembre 1999. [citée p. 31, 38, 64, 115]
- [57] A. GROSSMANN et J. MORLET : Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM Journal on Mathematical Analysis*, 15:723–736, 1984. [citée p. 29]
- [58] L. GUIBAS et J. STOLFI : Primitives for the manipulation of general subdivisions and the computation of voronoi. *ACM Transactions on Graphics*, 4:74–123, Avril 1985. [citée p. 26]
- [59] S. GUPTA, K. SENGUPTA et A. KASSIM : Registration and partitioningbased compression of 3d dynamic data. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(11):1144–1155, Octobre 2003. [citée p. 110]
- [60] S. GUPTA, K. SENGUPTA et A. KASSIM : 46789. *Comput. Vis. Image Underst.*, 87:116–130, Juillet 2002. [citée p. 104, 110]
- [61] I. GUSKOV : Manifold-based approach to semi-regular remeshing. *Graphical Models*, 69:1–18, Janvier 2007. [citée p. 31, 39, 56, 64, 115]

- [62] I. GUSKOV et A. KHODAKOVSKY : Wavelet compression of parametrically coherent mesh sequences. *In Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, Août 2004. [citée p. 104, 113]
- [63] I. GUSKOV, W. SWELDENS et P. SCHRÖDER : Multiresolution signal processing for meshes. *In Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 325–334, New York, Etats Unis, 1999. ACM. [citée p. 63]
- [64] I. GUSKOV, K. VIDIMCE, W. SWELDENS et P. SCHRÖDER : Normal meshes. *In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 95–102, New York, Etats Unis, 2000. ACM. [citée p. 31, 38, 39, 56, 57, 64, 115]
- [65] A. GUÉZIEC : Surface simplification inside a tolerance volume. Rapport technique, New York, Etats Unis, Mars 1996. IBM Research Report RC 20440. [citée p. 120]
- [66] H. HOPPE, T. DE ROSE, T. DUCHAMP, M. HALSTEAD, H. JIN, J. McDONALD, J. SCHWEITZER et W. STUETZLE : Piecewise smooth surface reconstruction. *In Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 295–302, New York, Etats Unis, 1994. ACM. [citée p. 22]
- [67] L. IBARRIA et J. ROSSIGNAC : Dynapack : Space-time compression of the 3d animations of triangle meshes with fixed connectivity. *ACM Symposium on Computer Animation*, pages 126–135, 2003. [citée p. 104, 112]
- [68] P. JINGLIANG, K. CHANG-SU et C.-C. JAY KUO : Technologies for 3d mesh compression : A survey. *Journal of Visual Communication and Image Representation*, 16:688–733, Décembre 2005. [citée p. 65]
- [69] P. JÖRG et R. ULRICH : The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics*, 16:420–431, Octobre 1997. [citée p. 22, 23]
- [70] A. KAMMOUN, F. PAYAN et M. ANTONINI : Adaptive semi-regular remeshing : A voronoi-based approach. *In Proceedings of IEEE International Workshop on MultiMedia Signal Processing*, Saint-Malo, France, Septembre 2010. [citée p. 28, 115]
- [71] A. KAMMOUN, F. PAYAN et M. ANTONINI : Approche adaptative basée voronoï pour le remaillage semi-régulier. *In Groupe de Recherche et d'Etudes du Traitement du Signal et des Images (GRETSI)*, Bordeaux, France, Septembre 2011. [citée p. 28]
- [72] A. KAMMOUN, F. PAYAN et M. ANTONINI : Optimisation du schéma lifting basé butterfly pour les maillages semi-réguliers. *In 13ème Edition d'ORASIS, Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur*, Praz-sur-Arly, France, Juillet 2011. [citée p. 31]
- [73] A. KAMMOUN, F. PAYAN et M. ANTONINI : Optimized butterfly-based lifting scheme for semi-regular meshes. *In Proceedings of IEEE International Conference in Image Processing (ICIP)*, Bruxelles, Belgique, Septembre 2011. [citée p. 31]
- [74] Z. KARNI et C. GOTSMAN : Spectral compression of mesh geometry. *In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 279–286, New York, Etats Unis, 2000. ACM Press. [citée p. 31]
- [75] Z. KARNI et C. GOTSMAN : Compression of soft-body animation sequences. *Computers & Graphics*, 28:25–34, 2004. [citée p. 104, 111, 112]

- [76] L. KETTNER : Using generic programming for designing a data structure for polyhedral surfaces. *Computational Geometry Theory and Applications*, 13:65–90, 1999. [citée p. 27]
- [77] A. KHODAKOVSKY et I. GUSKOV : Normal mesh compression. *Geometric Modeling for Scientific Visualization*, Springer-Verlag, 2002. [citée p. 64, 66, 104, 115]
- [78] A. KHODAKOVSKY, N. LITKE et P. SCHRÖDER : Globally smooth parameterizations with low distortion. *ACM Transactions on Graphics*, 22:350–357, 2003. [citée p. 38]
- [79] A. KHODAKOVSKY, P. SCHRÖDER et W. SWELDENS : Progressive geometry compression. In *SIGGRAPH : Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 271–278. ACM Press/Addison-Wesley Publishing Co., 2000. [citée p. 64, 65, 66, 84, 85, 104, 115, 130, 132, 137]
- [80] D. KING et J. ROSSIGNAC : Optimal bit allocation in 3d compression. *Journal of Computational Geometry, Theory and Applications*, 1999. [citée p. 31]
- [81] L. KOBBELT : Interpolatory subdivision on open quadrilateral nets with arbitrary topology. *Computer Graphics Forum*, 15:409–420, 1996. [citée p. 22, 23]
- [82] L.P. KOBBELT et M. BOTSCH : An interactive approach to point cloud triangulation. *Computer Graphics Forum*, 19, 2000. [citée p. 17]
- [83] L.P. KOBBELT, J. VORSATZ, U. LABSIK et H.-P. SEIDEL : A shrink wrapping approach to remeshing polygonal surfaces, 1999. [citée p. 31, 39, 64, 115]
- [84] J. KOVACEVIC et W. SWELDENS : Wavelet families of increasing order in arbitrary dimensions. *IEEE Transactions on Image Processing*, 9:480–496, Mars 2000. [citée p. 65]
- [85] J.C. LAGARIAS, J.A. REEDS, M.H. WRIGHT et P.E. WRIGHT : Convergence properties of the nelder-mead simplex algorithm in low dimensions. *SIAM Journal of Optimization*, 9:112–147, 1996. [citée p. 73, 79, 159]
- [86] S. LAVU, H. CHOI et R. BARANIUK : Geometry compression of normal meshes using rate-distortion algorithms. In *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, 2003. [citée p. 31, 64, 66, 104, 115]
- [87] A.W.F. LEE, W. SWELDENS, P. SCHRÖDER, L. COWSAR et D. DOBKIN : MAPS : Multiresolution adaptive parameterization of surfaces. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 95–104, New York, Etats Unis, 1998. ACM. [citée p. 31, 38, 39, 47, 48, 54, 56, 64, 115, 116, 119, 122, 143, 145]
- [88] K.-Y. LEE, S.-C. KANG et T.-W. KIM : Normal meshes for multiresolution analysis of irregular meshes with boundaries. *JSME International Journal Series C, Mechanical systems, Machine Elements and Manufacturing*, 45:628–636, Juin 2002. [citée p. 31, 38, 64, 115]
- [89] J.E. LENGUEL : Compression of time-dependent geometry. In *Proceedings of the 1999 symposium on Interactive 3D graphics*, I3D '99, pages 89–95, New York, NY, USA, 1999. ACM. [citée p. 104, 110]
- [90] D. LI, K. QIN et H. SUN : Unlifted loop subdivision wavelets. In *Proceedings of the Computer Graphics and Applications*, pages 25–33, Washington, Etats Unis, 2004. IEEE Computer Society. [citée p. 65]

- [91] J. LI, C. KUO et J. LI : Progressive compression of 3d graphic models. *In Proceedings of the 1997 International Conference on Multimedia Computing and Systems*, pages 135–142, Washington, Etats Unis, 1997. IEEE Computer Society. [citée p. 31]
- [92] Y. LINDE, A. BUZO et R. GRAY : An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28:84–94, 1980. [citée p. 46, 47, 54, 161]
- [93] S.P. LLOYD : Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137, Mars 1982. [citée p. 43, 117]
- [94] C. LOOP : Smooth subdivision surfaces based on triangles. Mémoire de D.E.A., University of Utah, 1987. [citée p. 22, 23, 24, 38, 53, 64, 79]
- [95] A.C. LOPES et M.N. GAMITO : Wavelet compression and transmission of deformable surfaces over networks. *In Proceedings of the 10th Portuguese Computer Graphics Meeting*, pages 107–114, 2001. [citée p. 104, 113, 129]
- [96] W.E. LORENSEN et H.E. CLINE : Marching cubes : A high resolution 3d surface construction algorithm. *SIGGRAPH Computer Graphics*, 21:163–169, Août 1987. [citée p. 18]
- [97] M. LOUNSBERY, T. DeROSE et J. WARREN : Multiresolution analysis for surfaces of arbitrary topological type. Mémoire de D.E.A., Department of Computer Science and Engineering, University of Washington, Octobre 1993. [citée p. 11, 31, 37, 64, 115]
- [98] M. LOUNSBERY, T. D. DeROSE et J. WARREN : Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics*, 16:34–73, Janvier 1997. [citée p. 64, 65]
- [99] S. MALLAT : A theory for multiresolution signal decomposition : the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989. [citée p. 29, 63]
- [100] K. MAMOU : *Compression de maillages 3D statiques et dynamiques*. Thèse de doctorat, Université Paris V - René Descartes, 2008. [citée p. 104, 111]
- [101] K. MAMOU, D. MARPE, T. ZAHARIA, M. PREDA et F. PRÊTEUX : Frame-based animated mesh compression : integration of the cabac arithmetic encoder. Rapport technique ISO/IEC JTC1/SC29/WG11, MPEG2007/M14493, San Jose, Etats unis, Avril 2007. [citée p. 111]
- [102] K. MAMOU, D. MARPE, T. ZAHARIA, M. PREDA et F. PRÊTEUX : Famc integration into the mpeg-4 refsoft. Rapport technique ISO/IEC JTC1/SC29/WG11, MPEG2008/M15150, Antalya, Turquie, Janvier 2008. [citée p. 111]
- [103] K. MAMOU, T. ZAHARIA et F. PRÊTEUX : Famc : The mpeg-4 standard for animated mesh compression. *In Proceedings of the International Conference on Image Processing (ICIP)*, pages 2676–2679, 2008. [citée p. 111]
- [104] K. MAMOU, T. ZAHARIA, F. PRÊTEUX, A. KAMMOUN, F. PAYAN et M. ANTONINI : Two optimizations of the mpeg-4 famc standard for enhanced compression of animated 3d meshes. *In Proceedings of IEEE International Conference in Image Processing (ICIP)*, San Diego, Etats Unis, Septembre 2008. [citée p. 111]
- [105] M. MANTYLA : *Introduction to Solid Modeling*. W. H. Freeman & Co., New York, Etats Unis, 1988. [citée p. 27]

- [106] M. MEYER, M. DESBRUN, P. SCHRÖDER et A. BARR : Discrete Differential Geometry Operators for Triangulated 2-Manifolds. *In International Workshop on Visualization and Mathematics*, 2002. [citée p. 48]
- [107] Y. MEYER : *Wavelets and operators*, volume 37 de *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1992. Translated from the 1990 French original by D.H. Salinger. [citée p. 29]
- [108] J. O'ROURKE : *Computational Geometry in C*. Cambridge University Press, New York, Etats Unis, 2nd édition, 1998. [citée p. 26]
- [109] H.M. OZAKTAS et L. ONURAL : *Three-Dimensional Television : Capture, Transmission, Display*. Springer Publishing Company, Incorporated, 1st édition, 2007. [citée p. 11, 107]
- [110] C. PAOLO, R. CLAUDIO et S. ROBERTO : Metro : measuring error on simplified surfaces. Rapport technique, Paris, France, 1996. [citée p. 31, 32]
- [111] F. PAYAN : *Optimisation du Compromis Débit-Distorsion pour la Compression Géométrique de Maillages Surfacciques Triangulaires*. Thèse de doctorat, Université de Nice-Sophia Antipolis, 2004. [citée p. 28, 30, 32, 133, 136]
- [112] F. PAYAN et M. ANTONINI : An efficient bit allocation for compressing normal meshes with an error-driven quantization. *Computer Aided Geometry Design*, 22:466–486, 2005. [citée p. 31, 64, 66, 84, 85, 86, 100, 104, 115, 135, 136, 146]
- [113] F. PAYAN et M. ANTONINI : Mean square error approximation for wavelet-based semiregular mesh compression. *IEEE Transactions on Visualization and Computer Graphics*, 12:649–657, Juillet 2006. [citée p. 64, 66, 85, 86, 100, 104, 115]
- [114] F. PAYAN et M. ANTONINI : Temporal wavelet-based geometry coder for 3d animations. *Computer & Graphics*, 31:77–88, Janvier 2007. [citée p. 129, 131, 136, 137, 141]
- [115] G. PIELLA et H.J.A.M. HEIJMANS : Adaptive lifting schemes with perfect reconstruction. *IEEE Transactions on Signal Processing*, 50:1620–1630, 2001. [citée p. 66, 67]
- [116] C. ROUDET, F. DUPONT et A. BASKURT : Multiresolution mesh segmentation based on surface roughness and wavelet analysis. *In Visual Communications and Image Processing*, volume 6508. SPIE, 2007. [citée p. 64]
- [117] C. ROUDET et F. PAYAN : Remaillage semi-régulier pour les maillages surfacciques triangulaires : un etats de l'art. *Revue Electronique Francophone d'Informatique Graphique*, 5(1):27–40, 2011. [citée p. 37, 39, 161]
- [118] M. SATTLER, R. SARLETTE et R. KLEIN : Simple and efficient compression of animation sequences. *In Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '05, pages 209–217, New York, Etats unis, 2005. ACM. [citée p. 112]
- [119] B. SAUVAGE, S. HAHMANN et G.P. BONNEAU : Volume preservation of multiresolution meshes. *Computer Graphics Forum*, 26, 2007. [citée p. 64]
- [120] S. SCHAEFER et J. WARREN : On C^2 triangle/quad subdivision. *ACM Transactions on Graphics*, 24:28–36, Janvier 2005. [citée p. 22]

- [121] P. SCHRÖDER et W. SWELDENS : Spherical wavelets : Efficiently representing functions on the sphere. *In Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 161–172, New York, Etats Unis, 1995. ACM. [citée p. 65]
- [122] A. SHAMIR et V. PASCUCCI : Temporal and spatial level of details for dynamic meshes. *In Proceedings of Virtual Reality Systems and Techniques*, pages 423–430, 2001. [citée p. 104, 112]
- [123] M.I. SHAMOS et D. HOEY : Closest-point problems. *In IEEE Symposium on Foundations of Computer Science*, pages 151–162, 1975. [citée p. 42]
- [124] J.-Y. SIM, C.-S. KIM, J.C.-C. KUO et S.-U. LEE : Normal mesh compression based on rate-distortion optimization. *In Proceedings of the International Workshop on MultiMedia Signal Processing*, Virgin Islands, Etats Unis, Décembre 2002. [citée p. 31, 64, 66, 104, 115]
- [125] J. STAM et C. LOOP : Quad/triangle subdivision, 2002. [citée p. 22]
- [126] N. STEFANOSKI et J. OSTERMANN : Connectivity-guided predictive compression of dynamic 3d meshes. *In Proceedings of IEEE International Conference in Image Processing (ICIP)*, Octobre 2006. [citée p. 104, 112]
- [127] E.J. STOLLNITZ, T. DEROSE et D.H. SALESI : *Wavelets for Computer Graphics : Theory and Applications*. Morgan Kaufmann Publishers Inc., 1996. [citée p. 11]
- [128] N. SUCIATI et K. HARADA : Wavelets-based multiresolution surface as framework for editing the global and local shapes. *International Journal of Computer Science and Network Security*, 2009. [citée p. 64]
- [129] W. SWELDENS : The lifting scheme : a custom-design construction of biorthogonal wavelets. *Applied and Computational Harmonic Analysis*, 3:186–200, 1996. [citée p. 65, 69, 131]
- [130] W. SWELDENS : The lifting scheme : a construction of second generation wavelets. *SIAM Journal on Mathematical Analysis*, 29:511–546, 1998. [citée p. 65, 69]
- [131] W. SWELDENS et P. SCHRÖDER : *Building Your Own Wavelets At Home*. SIGGRAPH Course Notes, 1996. [citée p. 69, 74]
- [132] W. SWELDENS et P. SCHRÖDER : *Digital Geometry Processing*. SIGGRAPH Course Notes, 2001. [citée p. 11, 35]
- [133] C. TOUMA et C. GOTSMAN : Triangle mesh compression. *Graphics Interface*, pages 26–34, 1998. [citée p. 85, 131]
- [134] B. USEVITCH : Optimal bit allocation for biorthogonal wavelet coding. *In Proceedings of the Conference on Data Compression*, Washington, Etats Unis, 1996. IEEE Computer Society. [citée p. 133]
- [135] S. VALETTE et R. PROST : Wavelet based multiresolution analysis of irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics*, 10:113–122, Mars 2004. [citée p. 53]
- [136] S. VALETTE et R. PROST : Wavelet-based progressive compression scheme for triangle meshes : Wavemesh. *IEEE Transactions on Visualization and Computer Graphics*, 10: 123–129, Mars 2004. [citée p. 63, 113]

- [137] L. VASA et V. SKALA : Coddyc : Connectivity driven dynamic mesh compression. *In 3DTV International Conference : True Vision- Capture, Transmission and Display of 3D Video*, 2007. [citée p. 104, 112]
- [138] L. VASA et V. SKALA : Geometry driven local neighbourhood based predictors for dynamic mesh compression. *Computer Graphics Forum*, 29(6):1921–1933, 2010. [citée p. 104, 112]
- [139] G. VORONOI : Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire : Recherches sur les paralléloèdres primitifs. *J. Reine Angew. Math*, 134:198–287, 1908. [citée p. 41, 42]
- [140] A. WITKIN : *Animation*. Computer Graphics I Course Notes, School of Computer Science, Carnegie-Mellon University, 1995. [citée p. 107, 108]
- [141] J.-H. YANG, C.-S. KIM et S.-U. LEE : Progressive compression of 3d dynamic sequences. *In IEEE International Conference on Image Processing*, pages 1975–1978, 2004. [citée p. 104, 114]
- [142] J.H. YANG, C.S. KIM, et S.U. LEE : Semi-regular representation and progressive compression of 3d dynamic mesh sequences. *IEEE Transactions on Image Processing*, 15:2531–2544, Septembre 2006. [citée p. 104, 114, 115]
- [143] J.H. YANG, C.S. KIM et S.-U. LEE : Compression of 3d triangle mesh sequences based on vertex-wise motion vector prediction. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(12):1178–1184, Décembre 2002. [citée p. 104, 112]
- [144] D. ZORIN, P. SCHRÖDER, T. DE ROSE, L. KOBELT, A. LEVIN et W. SWELDENS : Subdivision for modeling and animation. Course Notes, ACM SIGGRAPH, 2000. [citée p. 12]
- [145] D. ZORIN, P. SCHRÖDER et W. SWELDENS : Interpolating subdivision for meshes with arbitrary topology. *In Proceedings of the 23th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 189–192, New York, Etats Unis, Septembre 1996. ACM. [citée p. 22, 23, 29, 38, 70, 71, 72]
- [146] D. ZORIN, P. SCHRÖDER et W. SWELDENS : Interactive multiresolution mesh editing. *In Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 259–268, New York, Etats Unis, 1997. ACM. [citée p. 64, 68]

Appendices

Méthode de Nelder-Mead

La méthode de Nelder-Mead est un algorithme d'optimisation non-linéaire. Elle est publiée¹ par Nelder et Mead en 1965 [85]. C'est une méthode numérique qui minimise une fonction dans un espace à plusieurs dimensions. Cette méthode utilise le concept de simplex qui est un polytope de $N + 1$ sommets dans un espace à N dimensions. Soit N la dimension de l'espace où f prend ses valeurs. On démarre avec un simplex de cet espace. La première étape consiste à enlever le point du simplex où la fonction est maximale et à le remplacer par la réflexion de ce point par rapport au centre de gravité des N points restants. Si ce point est meilleur, on étire le simplex dans cette direction. Sinon, on est dans une vallée, et on réduit le simplex par une similitude centrée sur le point du simplex où la fonction est minimale.

Cette méthode est donnée par l'algorithme 3.

Algorithm 3 Méthode de Nelder-Mead

```

{initialiser un simplex :}
 $simplex = \{p_0, p_1, \dots, p_N\}$ 
repeat
   $p_h \leftarrow$  point du simplex pour lequel la fonction à la plus grande valeur
   $p_s \leftarrow$  point du simplex pour lequel la fonction à la deuxième plus grande valeur
   $p_l \leftarrow$  point du simplex pour lequel la fonction à la plus petite valeur
   $p_m \leftarrow$  point moyen du simplex sans considérer  $p_h$ 
  {Réflexion :}
   $p_r \leftarrow p_m + \alpha(p_m - p_h)$ 
  if  $f(p_r) < f(p_l)$  then
    {Expansion :}
     $p_e = p_m + \gamma(p_r - p_m)$ 
    if  $f(p_e) < f(p_r)$  then
       $p_h \leftarrow p_e$ 
    else
       $p_h \leftarrow p_r$ 
    end if
  else
    if  $f(p_r) \leq f(p_s)$  then
       $p_h \leftarrow p_r$ 
    else
      if  $f(p_r) < f(p_h)$  then
         $p_h \leftarrow p_r$ 
      end if
      {Contraction :}
       $p_c = p_m + \beta(p_h - p_m)$ 
      if  $f(p_c) > f(p_h)$  then
        for all  $p_i \in simplex$  do
           $p_i \leftarrow (p_i + p_l)/2$ 
        end for
      else
         $p_h \leftarrow p_c$ 
      end if
    end if
  end if
until Convergence

```

Table des figures

1	Extrait du film <i>Toy Story 3D</i> [2].	3
2	Extrait du jeux vidéo <i>Crysis</i> [3].	4
3	Extrait d'un simulateur chirurgical [4].	4
1	Schéma général de codage progressif pour les maillages surfaciques.	12
1.1	Différentes représentations de BUNNY.	16
1.2	Exemple de maillage polygonal.	19
1.3	Genre de quelques surfaces.	20
1.4	Exemple d'une représentation standard pour un maillage simple	21
1.5	Exemple d'une courbe de subdivision.	22
1.6	Exemple d'une surface de subdivision (images tirées de [19]).	23
1.7	Différents masques utilisés dans le schéma de subdivision Loop.	24
1.8	Exemple d'une représentation basée triangle.	25
1.9	Exemple d'une représentation basée arête (structure <i>winged-edge</i>).	26
1.10	Exemple d'une représentation basée demi-arête.	27
1.11	Structure d'un maillage semi-régulier.	29
1.12	Principe de l'analyse multirésolution.	30
1.13	Distance unilatérale entre deux surfaces.	32
2.1	Principe général du remaillage semi-régulier.	36
2.2	Exemple de paramétrisation de surface (image tirée de [117]).	37
2.3	Schéma général de notre méthode.	40
2.4	Décomposition de l'espaces en tourbillons par Descartes.	41
2.5	Exemple de diagramme de Voronoï en 2D.	43
2.6	Exemple de la relaxation de Lloyd.	45
2.7	Cellule duale d'un sommet v_i	45
2.8	Exemple de diagramme de Voronoï, qui est construit sur VENUS.	46
2.9	Impact de la masse ρ sur la construction d'un diagramme de Voronoï.	47
2.10	Différentes itérations de l'algorithme LBG [92] appliquées sur l'objet VENUS.	47
2.11	Simplification basé Voronoï de l'objet VENUS.	49
2.12	Principe d'une itération de notre étape de raffinement.	50

2.13	Etape de raffinement proposée.	51
2.14	Echantillonnage de l'objet VASELION.	52
2.15	Erreurs de remaillage pour l'objet en fonction du sur-échantillonnage.	54
2.16	Erreurs de remaillage pour VENUS en fonction du seuil de subdivision.	55
2.17	Erreurs de remaillage pour SKULL en fonction de la simplification utilisée.	55
2.18	Erreurs de remaillage pour BIMBA en fonction de la simplification utilisée.	56
2.19	L'objet SKULL remaillé avec notre méthode.	57
2.20	L'objet BIMBA remaillé avec notre méthode.	58
2.21	L'objet HORSE remaillé avec notre méthode.	58
2.22	L'objet RABBIT remaillé avec notre méthode.	58
2.23	L'objet VENUS remaillé avec notre méthode.	59
2.24	Courbes d'évolution de l'erreur de remaillage pour l'objet VENUS.	59
2.25	Courbes d'évolution de l'erreur de remaillage pour l'objet SKULL.	60
2.26	Courbes d'évolution de l'erreur de remaillage pour l'objet RABBIT.	60
2.27	Courbes d'évolution de l'erreur de remaillage pour l'objet HORSE.	61
3.1	Aperçu d'une décomposition en ondelettes pour maillages surfaciques.	64
3.2	Limites du schéma de subdivision de Butterfly.	67
3.3	Limites du schéma de subdivision de Loop.	68
3.4	Transformée en ondelettes d'un maillage semi-régulier.	69
3.5	Schéma <i>lifting</i> pour les maillages semi-réguliers : Analyse.	70
3.6	Schéma <i>lifting</i> pour les maillages semi-réguliers : Synthèse.	70
3.7	Masques utilisés lors de la prédiction du schéma <i>lifting</i> basé Butterfly.	71
3.8	Masque utilisé lors de la mise à jour du schéma <i>lifting</i> basé Butterfly.	74
3.9	Méthode de calcul de la valeur du paramètre γ	74
3.10	Schéma <i>lifting</i> basé Loop proposé dans [15].	77
3.11	Masque utilisé lors de l'étape de prédiction du schéma <i>lifting</i> basé Loop.	78
3.12	Masques utilisés lors de la mise à jour du schéma <i>lifting</i> basé Loop.	78
3.13	Fonctions d'échelle et fonction d'ondelettes de la transformée basée Loop.	81
3.14	Fonction d'échelle ϕ_1 et fonction d'ondelettes ψ_1	81
3.15	Principe général d'un codeur géométrique.	84
3.16	PSNR pour VASE LION avec le schéma basé Butterfly et le codeur PGC.	90
3.17	PSNR pour VASE LION avec le schéma basé Butterfly et le codeur EDQ.	90
3.18	Distribution de l'erreur géométrique pour VASE LION.	94
3.19	Comparaison visuelle pour KNOT.	95
3.20	PSNR pour VASE LION avec le schéma basé Loop et le codeur PGC.	96
3.21	Différentes résolutions relatives à la transformée basée Loop.	97
3.22	Différentes résolutions relatives à la transformée basée Loop (zoom).	98
1	Schéma de compression proposé pour les séquences de maillages surfaciques.	105
2	Séquence de maillages surfaciques CHICKEN.	109
3	Exemple d'animations par images-clés.	109

4	Exemple de structure avec des <i>I-frames/P-frames/B-frames</i>	113
5.1	Notre approche pour le remaillage semi-régulier des séquences de maillages.	116
5.2	Segmentation de la séquence FLYING HORSE.	118
5.3	Recalage du maillage M_{44} sur le maillage M_1	119
5.4	Exemple de contraction entraînant un changement topologique.	120
5.5	Exemple de contraction entraînant une inversion de la normale.	121
5.6	Exemple de contraction entraînant un triangle compacte.	121
5.7	L'objet DOLPHIN original et simplifié.	121
5.8	Séquence DOLPHIN remaillée.	124
5.9	Séquence FLYING VENUS remaillée.	125
5.10	Séquence FLYING HORSE remaillée.	126
6.1	Schéma de compression proposé.	130
6.2	Le mouvement de chaque sommet est considéré comme un signal 1D.	131
6.3	Décomposition spatio-temporelle basée sur deux transformées en ondelettes.	132
6.4	Schéma de codage basé sur un seul niveau de décomposition temporelle.	134
6.5	Distributions typiques des coefficients d'ondelettes.	136
6.6	Courbes d'évolution débit-distorsion pour la séquence DOLPHIN.	138
6.7	Courbes d'évolution débit-distorsion pour la séquence FLYING VENUS.	138
6.8	Courbes d'évolution débit-distorsion pour la séquence FLYING HORSE.	139
6.9	Courbes d'évolution distorsion-trame pour la séquence DOLPHIN.	139
6.10	Courbes d'évolution distorsion-trame pour la séquence FLYING VENUS.	140
6.11	Courbes d'évolution distorsion-trame pour la séquence FLYING HORSE.	140

Liste des tableaux

1.1	Classification des principaux schémas de subdivision [44].	23
2.1	Caractéristiques des différents objets utilisés.	53
3.1	Valeurs des 43 sommets non nuls de V^j obtenus après une étape de synthèse.	75
3.2	Normes $L1$ des coefficients d'ondelettes (Butterfly) pour l'objet VASE LION. .	76
3.3	Normes $L1$ des coefficients d'ondelettes (Butterfly) pour l'objet JOAN OF ARC.	76
3.4	Normes $L1$ des coefficients d'ondelettes (Butterfly) pour l'objet BIMBA . . .	77
3.5	Normes $L1$ des coefficients d'ondelettes (Loop) pour l'objet VASE LION. . .	83
3.6	Normes $L1$ des coefficients d'ondelettes (Loop) pour l'objet JOAN OF ARC. .	83
3.7	Normes $L1$ des coefficients d'ondelettes (Loop) pour l'objet BIMBA.	83
3.8	Résultats de compression pour VASE LION avec Butterfly et PGC.	87
3.9	Résultats de compression pour BIMBA avec Butterfly et PGC.	87
3.10	Résultats de compression pour RABBIT avec Butterfly et PGC.	88
3.11	Résultats de compression pour VASE LION avec Butterfly et EDQ.	88
3.12	Résultats de compression pour BIMBA avec Butterfly et EDQ.	89
3.13	Résultats de compression pour RABBIT avec Butterfly et EDQ.	89
3.14	Temps de calcul pour l'optimisation du schéma basé Butterfly.	91
3.15	Résultats de compression pour VASE LION avec Loop et PGC.	92
3.16	Résultats de compression pour BIMBA avec Loop et PGC.	92
3.17	Résultats de compression pour RABBIT avec Loop et PGC.	93
5.1	Caractéristiques des séquences utilisées durant les expérimentations.	123
5.2	Temps de calcul nécessaire pour remailler la séquence DOLPHIN.	127

Publications

Journal international avec comité de lecture

- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Sparsity-based optimization of two lifting-based wavelet transforms for semi-regular mesh compression", *Computer & Graphics*, Elsevier, Juillet, 2011, (Article soumis).

Conférences internationales avec comité de lecture

- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Optimized Butterfly-based lifting scheme for semi-regular meshes", *IEEE International Conference on Image Processing (ICIP)* Bruxelles, Belgique, Septembre, 2011.
- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Adaptive Semi-Regular Remeshing : A Voronoi-Based Approach", *IEEE International Workshop on MultiMedia Signal Processing (MMSP)*, Saint Malo, France, Octobre 2010.
- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Bit allocation for Spatio-temporal Wavelet Coding of Animated Semi-regular Meshes", *Proceedings of 15th International MultiMedia Modeling Conference (MMM)*, Sophia Antipolis, France, Janvier, 2009.
- Khaled MAMMOU, Titus ZAHARIA, Françoise PRETEUX, Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Two Optimizations of the MPEG-4 FMC standard for Enhanced Compression of Animated 3D Meshes", *IEEE International Conference on Image Processing (ICIP)*, San Diego, Etats unis, Octobre, 2008.
- Frédéric PAYAN, Aymen KAMMOUN et Marc ANTONINI, "Remeshing and spatio-temporal wavelet filtering for 3D animations", *IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP)*, Las Vegas, Etats unis, Mars, 2008.

Conférences nationales avec comité de lecture

- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Approche adaptative basée Voronoï pour le remaillage semi-régulier", Groupe de Recherche et d'Etudes du Traitement du Signal et des Images (GRETSI), Bordeaux, France, Septembre, 2011.
- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Optimisation du schéma lifting basé Butterfly pour les maillages semi-réguliers", Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS), Praz-sur-Arly, France, Juillet, 2011.

Rapports de recherche

- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Adaptive Semi-Regular Remeshing : A Voronoi-Based Approach", I3S Research Report, I3S/RR-2010-07-FR, Mai, 2010.
- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "A Feature-Preserving Remeshing Scheme For Surface Meshes", I3S Research Report, I3S/RR-2009-03-FR, Mars, 2009.
- Aymen KAMMOUN, Frédéric PAYAN et Marc ANTONINI, "Remaillage et analyse spatio-temporelle basée ondelettes pour les animations 3D", I3S Research Report, I3S/RR-2008-02-FR, Janvier, 2008.